



UNIVERSIDAD
NACIONAL
DE COLOMBIA

**DESARROLLO DE UN AMBIENTE DE SIMULACIÓN PARA
TOPOLOGÍAS DE BUS DE PROCESO MULTI-
PROVEEDOR, APLICADO A LAS SUBESTACIONES DEL
SISTEMA INTERCONECTADO NACIONAL DE COLOMBIA**

Francisco Javier Serna Alzate

Universidad Nacional de Colombia

Facultad de Minas

Medellín, Colombia

2017

DESARROLLO DE UN AMBIENTE DE SIMULACIÓN PARA TOPOLOGÍAS DE BUS DE PROCESO MULTI- PROVEEDOR, APLICADO A LAS SUBESTACIONES DEL SISTEMA INTERCONECTADO NACIONAL DE COLOMBIA

Francisco Javier Serna Alzate

Tesis o trabajo de investigación presentada(o) como requisito parcial para optar al título
de:

Magister en Ingeniería – Ingeniería Eléctrica

Director (a):

PhD. Germán Darío Zapata Madrigal

Línea de Investigación:

Comunicaciones en Sistemas Eléctricos

Universidad Nacional de Colombia

Facultad de Minas

Medellín, Colombia

2017

Agradecimientos

El autor agradece al PhD. Germán Darío Zapata Madrigal, profesor de la Universidad Nacional de Colombia – Sede Medellín y Coordinador del Grupo de Teleinformática y Teleautomática T&T de la Universidad Nacional, por sus valiosas orientaciones durante el desarrollo del presente trabajo. También se agradece al ingeniero Rafael Santamaría Isaza, Coordinador Técnico del Laboratorio IEC 61850 de la Universidad Nacional de Colombia – Sede Medellín, por sus valiosos aportes durante la validación de los resultados del presente trabajo.

Resumen

Este trabajo presenta el desarrollo de un ambiente de simulación para arquitecturas de bus de proceso, con un enfoque multi-proveedor y aplicable a las subestaciones del Sistema Interconectado Nacional – SIN. Dicho desarrollo comprendió la recopilación de parámetros reales de desempeño de dispositivos comerciales, de cinco proveedores diferentes, así como el análisis de una muestra de 20 subestaciones de 110 a 220kV. Adicionalmente, se analizó una muestra de 10 subestaciones de 500kV del SIN, para determinar las topologías de bus de proceso aplicables según la norma IEC 61850. Toda esta información fue integrada en un simulador de bus de proceso implementado sobre OMNET++, el cual permite construir topologías de bus de proceso con redundancia PRP a partir de distintas referencias comerciales de dispositivos.

Mediante la aplicación del ambiente de simulación desarrollado, fue posible determinar las tasas de pérdida de paquetes, los retardos y la influencia de cada dispositivo comercial en el desempeño de las arquitecturas de bus de proceso.

Se espera que este trabajo resulte útil en la planeación y validación de arquitecturas de bus de proceso en las subestaciones del SIN colombiano.

Palabras clave: ambiente de simulación, bus de proceso, IEC 61850, dispositivos comerciales, multi-proveedor, desempeño de red de comunicaciones

Abstract

This work presents the development of a simulation environment for process bus architectures, focused on multi-vendor technologies and suitable for the Colombian National Interconnected System – SIN. Such development comprised the gathering of real performance parameters of commercial devices, provided by five different vendors. Furthermore, a sample of 20 substations at 110-220kV and 10 substations at 500kV from the SIN was analyzed, to determine the suitable bus process architectures, according to the standard IEC 61850. All this information was integrated in a process bus simulator, programmed under OMNET++. This simulator is able to build process bus topologies with PRP redundancy, taking into account the commercial references for different devices. By using such simulator, it was possible to determine the packet-loss rates, transmission delay and the influence of each commercial device on the performance of process bus architectures.

It is looked forward that this work be useful to plan and validate process bus architectures for the Colombian SIN substations.

Keywords: simulation environment, process bus, IEC 61850, commercial devices, multi-vendor, communication network performance

Contenido

	Pág.
1. Planteamiento del problema.....	5
2. Justificación	8
3. Objetivos.....	10
4. Metodología.....	11
5. Marco Teórico.....	13
6. Estado del arte.....	23
7. Estimación de índices de desempeño para dispositivos comerciales.....	28
8. Caracterización de requerimientos de desempeño del bus de proceso para una subestación típica del SIN colombiano	44
9. Desarrollo de la aplicación en OMNET++	59
10. Validación de resultados obtenidos	98
11. Conclusiones y recomendaciones.....	107
12. Referencias bibliográficas.....	111
Anexo A: Correspondencia intercambiada con fabricantes durante la búsqueda de información primaria.....	123
Anexo B: Diagramas unifilares de las subestaciones analizadas	148
Anexo C: Guía para construcción de arquitecturas de bus de proceso en OMNET++	158
Registro de información en las simulaciones	167
Ingreso de parámetros por parte del usuario.....	172
Conexión de múltiples dispositivos a pedido del usuario	174
Referencias bibliográficas	176

Lista de figuras

Figura 5 - 1: Posibles arquitecturas del bus de proceso. a) Punto a punto; b) Estrella dual; c) anillo simple; d) anillo dual; e) estrella con PRP (Mo et al., 2010)	15
Figura 5 - 2: Topología Cobweb (X. Liu et al., 2014)	16
Figura 5 - 3: Topología Cobweb aplicada al bus de proceso (X. Liu et al., 2014)	16
Figura 5 - 4: Arquitectura de comunicación entre controladores de bahía (Tournier & Werner, 2010)	17
Figura 5 - 5: Tiempos de procesamiento de datos según la norma IEC 61850-5 (Van Rensburg et al., 2015).....	21
Figura 8 - 1: Diagrama unifilar del Sistema de Transmisión Nacional. Fuente: (UPME, 2016)	46
Figura 8 - 2: Diagrama unifilar de las subestación Bolívar 220kV (UPME, 2010)	47
Figura 8 - 3: Diagrama unifilar de la subestación Chinú 500kV (UPME, 2013)	48
Figura 8 - 4: Configuraciones de las subestaciones de 500kV (elaboración propia)	50
Figura 8 - 5: Configuraciones de las subestaciones 110-220-230kV (elaboración propia)	51
Figura 8 - 6: Diagrama unifilar de la subestación Sonsón (EPM, Anexo B)	54
Figura 8 - 7: Ejemplo de topología empleada por EPM (EPM, Anexo B)	55
Figura 8 - 8: Arquitectura estándar PRP por bahía (Tournier & Werner, 2010)	57
Figura 8 - 9: Arquitectura estándar HSR por bahía (Tournier & Werner, 2010)	57
Figura 9 - 1: Arquitectura sugerida por el Laboratorio IEC 61850 – UNAL (elaboración propia, según orientación del Laboratorio IEC 61850 UNAL)	62
Figura 9 - 2: Listado de equipos usados en la arquitectura (elaboración propia)	63
Figura 9 - 3: Arquitectura genérica implementada en OMNET++ (elaboración propia)....	64
Figura 9 - 4: Arquitectura con una MU en cada rama (elaboración propia).....	66
Figura 9 - 5: Arquitectura con diversas MU en cada rama (elaboración propia)	66
Figura 9 - 6: Dinámica del tráfico de red en el bus de proceso (elaboración propia según orientación del Laboratorio IEC 61850 UNAL)	68
Figura 9 - 7: Cuadros de diálogo de configuración del usuario (elaboración propia)	81
Figura 9 - 8: Topología resultante de la configuración del usuario (elaboración propia) ..	81
Figura 9 - 9: Configuración de filtrado VLAN por el usuario (elaboración propia)	82
Figura 9 - 10: Filtrado VLAN desactivado (elaboración propia)	83
Figura 9 - 11: Filtrado VLAN activado (elaboración propia)	83

Figura 9 - 12: Mensajes de solicitud de suscripción a SV (elaboración propia).....	84
Figura 9 - 13: Conteo de paquetes enviados y recibidos (elaboración propia).....	86
Figura 9 - 14: Resultados de los retardos en algunos IEDs (elaboración propia).....	88
Figura 9 - 15: Representación gráfica del enlace defectuoso (elaboración propia)	95
Figura 10 - 1: Captura del tráfico sobre la LAN A – Reason MU320 (suministrada por Laboratorio IEC 61850 UNAL).....	99
Figura 10 - 2: Captura de tráfico sobre LAN B – Reason MU320 (suministrada por Laboratorio IEC 61850 UNAL).....	99
Figura 10 - 3: Detalle del contenido del paquete <i>Sampled Value</i> (suministrada por Laboratorio IEC 61850 UNAL).....	100
Figura 10 - 4: Simulación de captura de tráfico en OMNET++ para la Reason MU320 (elaboración propia)	101
Figura 10 - 5: Contenido del paquete <i>Sampled Value</i> en OMNET++ (elaboración propia)	102
Figura 10 - 6: Tráfico capturado por OMNET++ para la Reason MU320 (elaboración propia).....	102
Figura 10 - 7: Captura de tráfico LAN A – SIPROTEC 6MU850 (suministrada por Laboratorio IEC 61850 UNAL).....	104
Figura 10 - 8: Captura de tráfico LAN B – SIPROTEC 6MU850 (suministrada por Laboratorio IEC 61850 UNAL).....	104
Figura 10 - 9: Captura de tráfico en OMNET++ para la SIPROTEC 6MU805	106

Lista de tablas

Tabla 5 - 1: Requerimientos de QoS para pérdida de paquetes (Sugeng, Istiyanto, Mustofa, & Ashari, 2015).....	18
Tabla 5 - 2: Requerimientos de QoS para el retardo (Sugeng et al., 2015)	18
Tabla 5 - 3: Requerimientos de QoS para el Jitter (Sugeng et al., 2015).....	18
Tabla 5 - 4: Requerimientos de QoS para el throughput (Sugeng et al., 2015)	19
Tabla 5 - 5: Índices MTTF y MTTR para algunos componentes del bus de proceso (Liu, Panteli, & Crossley, 2014).....	20
Tabla 5 - 6: Requerimientos de tiempos de transferencia según IEC/TR 61850-90-4 (IEC, 2013)	21
 Tabla 7 - 1: Parámetros de los IED de control (elaboración propia)	31
Tabla 7 - 2: Parámetros identificados para IEDs de Protección (elaboración propia)	36
Tabla 7 - 3: Datos técnicos de diferentes referencias MiCOM (Alstom, 2012a)	38
Tabla 7 - 4: Parámetros de desempeño para <i>switches</i> (elaboración propia)	39
Tabla 7 - 5: Parámetros de red para las <i>Merging Units</i> (elaboración propia).....	41
 Tabla 8 - 1: Configuraciones y circuitos en subestaciones de 500kV (elaboración propia)	49
Tabla 8 - 2: Configuraciones y circuitos en subestaciones 110-220-230kV (elaboración propia)	50
Tabla 8 - 3: Parámetros de desempeño de comunicación según resolución CREG 025 de 1995 (CREG, 1995).....	53
 Tabla 9 - 1: Resultados de simulación para el caso 1 (elaboración propia)	89
Tabla 9 - 2: Resultados del caso 2 (elaboración propia).....	89
Tabla 9 - 3: Resultados del caso 3 (elaboración propia).....	91
Tabla 9 - 4: Resultados del caso 4 para la MU Genérica (elaboración propia)	92
Tabla 9 - 5: Resultados del caso 4 para la Reason MU320 (elaboración propia)	92
Tabla 9 - 6: Resultados del caso 4 para la SAM600CT y SAM600VT(elaboración propia)	92
Tabla 9 - 7: Resultados del caso 4 para la SIPROTEC6MU805 (elaboración propia).....	93
Tabla 9 - 8: Resultados del caso 4 para la AMU VIZIMAX (elaboración propia)	93
Tabla 9 - 9: Pérdida de enlace con la MU genérica (elaboración propia).....	95
Tabla 9 - 10: Pérdida de enlace con la Reason MU320 (elaboración propia)	95

Tabla 9 - 11: Pérdida de enlace con la SAM600CT y SAM600VT (elaboración propia) ..	95
Tabla 9 - 12: Pérdida de enlace con la SIPROTEC 6MU805 (elaboración propia).....	96
Tabla 9 - 13: Pérdida de enlace con la AMU VIZIMAX (elaboración propia)	96
Tabla 10 - 1: Tasa de muestreo de la Reason MU320.....	100
Tabla 10 - 2: Tasa de muestreo según OMNET++	103
Tabla 10 - 3: Tasa de muestreo para la SIPROTEC 6MU805.....	105
Tabla 10 - 4: Tasa de muestreo según OMNET++ para la SIPROTEC 6MU805	106

Introducción

Las subestaciones de potencia constituyen la realización física de los nodos del sistema eléctrico y juegan un papel de vital importancia en la operación y control del mismo. Algunas de las tareas de operación y control son realizadas a nivel local, sobre los equipos que componen la subestación misma, como los transformadores de tensión y corriente, los interruptores, seccionadores y los dispositivos electrónicos inteligentes (IEDs) de control y protección. Estas tareas se conocen como nivel de proceso y se conocen como tareas de nivel de proceso.

Estas tareas de nivel de proceso, que tradicionalmente se había implementado mediante cableado físico con altos costos de implementación y mantenimiento, constituyen un campo de investigación cada vez más prolífico, tras la publicación del estándar IEC 61850. Este estándar proporciona el marco de referencia para la automatización de subestaciones de potencia, mediante la incorporación de tecnologías de información y comunicación en los sistemas de monitoreo y control automático.

La investigación reciente ha abordado cuestiones como la configuración de los sistemas de comunicación que permitan la realización del bus de proceso bajo la norma IEC 61850 (Kasztenny, McGin, & Adamiak, 2009), (Tan & Luan, 2011), así como la evaluación del desempeño de arquitecturas concretas del bus de proceso (Tournier & Werner, 2010), (Mo, Tan, Crossley, Bo, & Klimek, 2010). Más aún, algunos trabajos se han centrado en simular el desempeño de alguno de los servicios concretos propuestos por la norma IEC 61850, como mensajes GOOSE o sincronización *Precisión Time Protocol – PTP* (Sidhu, Kanabar, Injeti, & Parikh, 2010), (Giorgi & Narduzzi, 2007).

Sin embargo, los trabajos mencionados evalúan el desempeño del bus de proceso a partir de índices generales, sin entrar en detalles sobre el rendimiento de dispositivos reales. Este punto es importante, ya que la interoperabilidad es uno de los temas que más preocupa a los operadores de las subestaciones actualmente. Por otro lado, es difícil encontrar trabajos que integren todos los servicios de la norma IEC 61850 y que además estén adaptados al contexto del Sistema de Interconectado Nacional – SIN.

Por estas razones, el presente trabajo propone un ambiente de simulación para arquitecturas de bus de proceso, con orientación multi-proveedor y aplicable a las subestaciones del SIN. Este ambiente de simulación permite configurar, a voluntad del usuario, topologías de bus de proceso con redundancia *Parallel Redundancy Protocol – PRP*, modificando parámetros como el número de dispositivos, el filtrado VLAN en los *switches* de red o la suscripción a diferentes servicios como los mensajes *Sampled Value* o *GOOSE*.

Para la construcción de este simulador fue necesario recopilar información en dos frentes principales: en primer lugar, se consultó una amplia variedad de catálogos de *Merging Units*, IEDs de control y protección y *switches*, para caracterizar los parámetros de desempeño de los dispositivos comerciales. En segundo lugar, se caracterizó también una amplia muestra de subestaciones del SIN, con el fin de determinar las arquitecturas de bus de proceso que son aplicables y recomendables para implementar en el simulador. El ambiente de simulación fue desarrollado en la herramienta OMNET++, de libre acceso para la comunidad académica.

Este trabajo está organizado de la siguiente manera: los primeros capítulos presentan el planteamiento del problema y su relevancia, así como los objetivos que se pretendieron alcanzar mediante el desarrollo del ambiente de simulación. Luego se expone el marco teórico del bus de proceso y el estado del arte de la investigación sobre este tema. Posteriormente, se expone todo el trabajo de caracterización de los índices de desempeño de los dispositivos comerciales y la caracterización de las subestaciones del SIN y las topologías de bus de proceso recomendables para el caso colombiano. Finalmente, se muestran los pasos que se siguieron para desarrollar el simulador, así como algunos ejemplos de simulaciones y validaciones en el laboratorio IEC 61850 de la Universidad Nacional de Colombia.

Se espera que este trabajo pueda ser útil para futuras investigaciones relacionadas con la planeación y evaluación del bus de proceso en subestaciones del SIN colombiano, ya que el simulador permite evaluar parámetros de desempeño como la tasa de pérdida de paquetes, los retardos y la sensibilidad de la red ante cambios en el número o referencia de los dispositivos del bus de proceso.

1. Planteamiento del problema

El estándar IEC 61850 tiene por objeto garantizar la interoperabilidad de los diferentes equipos y dispositivos involucrados en la automatización de subestaciones de potencia. Particularmente, el apartado IEC 61850-9-2 define los requisitos que deben cumplir los valores muestreados (*Sampled Values – SV*) que son tomados de los dispositivos de medición como transformadores de tensión y corriente y que son llevados a las unidades de captura de datos (*Merging Units – MU*) para su posterior acceso por un IED. Estos dispositivos, junto con la red LAN empleada para comunicarlos, constituyen lo que se llama bus de proceso, que además es empleado para controlar equipos de alta tensión, como interruptores y *switches* (Skendzic, Ender, & Zweigle, 2007).

Por otro lado, la implementación del bus de proceso supone diversas restricciones opuestas (Tournier & Werner, 2010): por ejemplo, el bus de proceso requiere ser escalable, pero los tiempos de retardo completos no deben superar los 3ms; por otro lado, se requieren altos índices de confiabilidad y disponibilidad del bus de proceso, pero a su vez se requiere una implementación cuyos costos no sean muy elevados. Por estas razones, la implementación del bus de proceso constituye todavía un desafío para un amplio número de aplicaciones.

Dadas las restricciones opuestas que se presentan en términos de tiempos de transmisión, confiabilidad, disponibilidad y costos, diversos trabajos se han propuesto como objetivo determinar el desempeño de las diferentes arquitecturas de bus de proceso.

Por ejemplo, (Mo et al., 2010) describe los principales tipos de arquitecturas; (Tournier & Werner, 2010) define los criterios para evaluar el desempeño de las diversas arquitecturas; (X. Liu, Pang, Zhabg, & Xu, 2014) utiliza índices *Mean Time To Repair* – MTTR y *Mean Time To Failure* – MTTF para determinar el desempeño de una arquitectura concreta; (Mitalkumar, 2011) también tiene en cuenta los índices MTTR y MTTF para determinar la confiabilidad combinatoria de las arquitecturas del bus de proceso en sistemas de protección.

El índice MTTF se refiere al tiempo promedio que tarda un sistema en fallar, también es conocido como MTBF (*Mean Time Before Failure*). Por su parte, el índice MTTR corresponde al tiempo promedio que transcurre entre la ocurrencia de una falla en el sistema y el momento en el que se restablece su servicio tras la reparación del mismo.

Sin embargo, los trabajos anteriores presentan tablas y estimaciones de los índices MTTF y MTTR de manera muy general: presentan los índices para un IED, una *Merging Units* (MU) o un *Switch Ethernet*, pero no se hace ninguna discriminación con respecto a otras características de cada dispositivo. Así por ejemplo, podría esperarse que el MTTF no sea exactamente igual para un IED costoso que para uno más barato, o para un *Switch Ethernet* de 8 puertos que para uno de 24 puertos.

Es posible que variables como el costo, la carga de uso, el tiempo de servicio (o vida útil remanente), e incluso la compañía fabricante, constituyan factores que afectan los índices MTTR y MTTF para cada dispositivo del bus de proceso.

Por estas razones, se hace necesario desarrollar un análisis de la arquitectura óptima del bus de proceso más orientado a la práctica, a la implementación real en subestaciones reales, que tenga en cuenta el desempeño de cada componente según sus características de operación y según el fabricante que haya desarrollado cada dispositivo. Se plantean entonces las siguientes preguntas de investigación:

- ¿De qué manera influyen factores como el fabricante, el costo y las condiciones de operación, en los índices de desempeño MTTF y MTTR para cada componente del bus de proceso?
- ¿Cuáles características de una subestación real del Sistema Interconectado Nacional (SIN) influyen en la elección de la arquitectura del bus de proceso y sus componentes?

- ¿Cómo determinar la configuración óptima para el bus de proceso de una subestación real, a partir de características concretas de los diversos dispositivos disponibles comercialmente y fabricados por distintos proveedores?

En resumen, el problema consiste en que no se cuenta con una metodología para determinar cuáles dispositivos deberían emplearse en el bus de proceso de una subestación real del SIN, que permita seleccionar entre las diversas opciones comerciales y los diversos fabricantes, ya que la investigación reciente se ha enfocado en índices de falla MTTF y MTTR muy generales.

Con el objetivo de responder estas preguntas de investigación, se plantea una caracterización de los dispositivos comerciales susceptibles de emplearse en el bus de proceso, para determinar índices de desempeño por dispositivo más detallados. Estos índices serían empleados luego en un aplicativo que integre el análisis de bloques de confiabilidad *Reliability Block Diagram – RBD*: cada bloque representa la confiabilidad de un dispositivo y se efectúan operaciones sobre estos bloques para obtener la confiabilidad conjunta de dispositivos conectados en serie y/o paralelo. Con esto, se ejecutan simulaciones combinatorias de múltiples topologías de bus de proceso en OMNET++, teniendo en cuenta parámetros reales de las subestaciones del SIN colombiano, tales como número de bahías, número de transformadores y nivel de tensión. Finalmente, los resultados de la optimización de la topología se compararían con los obtenidos por otro método descrito en la literatura, con el fin de validar la precisión del método propuesto.

2. Justificación

El desempeño del bus de proceso ha sido un problema tratado por diversos autores en la literatura desde hace relativamente poco tiempo. Así por ejemplo, (N. Liu, Panteli, & Crossley, 2014) emplea un análisis de bloques de confiabilidad RBD y simulaciones estocásticas con análisis de Monte Carlo, para determinar la confiabilidad de múltiples topologías de un bus de proceso. Para ello, se basa en los índices MTTR y MTTF de los equipos que componen el bus de proceso, pero estos índices son muy generales y no se relacionan con referencias comerciales disponibles en el mercado.

Otro trabajo notable es (Kumar, Das, & Islam, 2015), que emplea la herramienta OPNET para simular el comportamiento de una topología de red 61850-9-2 para una subestación de 132kV. Se obtienen detalles sobre los retardos en mensajes GOOSE que se producen en una topología anillo. Sin embargo, este trabajo no compara el desempeño de otras topologías ni ofrece detalles sobre los parámetros de desempeño particulares de los equipos de la subestación.

Un trabajo más antiguo es (Hodder, Kaszteny, McGin, & Hunt, 2009), que presenta las ventajas económicas de la implementación de un bus de proceso, en términos de ahorro de cableado y mano de obra. Este trabajo no presenta relación entre el costo de implementación de un bus de proceso determinado y su desempeño.

Otro ejemplo de evaluación del bus de proceso es (Yang, Yang, Ahn, & Kim, 2012), que va un paso más allá, realizando el diseño y la simulación de una red de comunicaciones IEC 61850 para la subestación Poong-Dong, declarada como la primera en implementar tecnología de múltiples proveedores. Sin embargo, en este trabajo sólo se toma una topología anillo con *High-availability Seamless Redundancy* – HSR y la evaluación del desempeño de dicha topología considera únicamente el tiempo de retardo promedio.

En cuanto a la experiencia de Colombia en este tema, la primera implementación real de un bus de proceso en Colombia corresponde a un proyecto desarrollado por HMV Ingenieros en la subestación de salida de 110kV de la Central Hidroeléctrica San Miguel, de 42 MW (Franco, García, Osorno, & Latozefski, 2015). No obstante, en este trabajo se evidencia que las *Merging Units* y los controladores de bahía empleados en el bus de proceso son todos fabricados por Alstom. Se infiere entonces, que no se cuenta con experiencias en Colombia de la implementación de un bus de proceso cuyos elementos sean suministrados por múltiples proveedores.

En conclusión, se puede afirmar que si bien el desempeño de las topologías de bus de proceso ha sido objeto de estudio desde hace cierto tiempo, los trabajos que comparan el desempeño de varias topologías no tienen en cuenta detalles técnicos de dispositivos disponibles en el mercado, sino aproximaciones generales de índices como el MTTR y MTTF. Adicionalmente, buena parte de las experiencias de implementación de bus de proceso, tanto en Colombia como en el mundo, se basan en soluciones suministradas enteramente por un único proveedor, limitando así de cierta forma el concepto de interoperabilidad promovido por el estándar IEC 61850. Se considera entonces pertinente desarrollar un estudio que tenga en cuenta información detallada de dispositivos disponibles en el mercado y fabricados por distintos proveedores, de manera que se pueda contar con insumos para diseñar subestaciones inteligentes más flexibles en Colombia. La presente propuesta de investigación contempla el desarrollo de un ambiente de simulación, que permita integrar información técnica de dispositivos comerciales de múltiples fabricantes con las exigencias de subestaciones reales del Sistema Interconectado Nacional.

3. Objetivos

3.1 Objetivo general

Desarrollar un ambiente de simulación para arquitecturas de bus de proceso multi-proveedor, mediante la integración de información de dispositivos comerciales en OMNET++ y la validación de resultados en laboratorio, para evaluar el desempeño de posibles arquitecturas y reducir incertidumbres en la implementación de subestaciones inteligentes en el Sistema Interconectado Nacional Colombiano.

3.2 Objetivos específicos

- Estimar índices de desempeño para dispositivos comerciales empleados en la realización del bus de proceso, mediante la caracterización técnica de dispositivos fabricados por al menos tres proveedores, para determinar la relación del desempeño con criterios como costo, fabricante y condiciones de operación.
- Caracterizar los requerimientos de desempeño del bus de proceso para una subestación típica del SIN colombiano, mediante el análisis de las topologías de subestación y las restricciones de las funciones de protección implementadas en ella.
- Integrar la información recopilada en una aplicación desarrollada en OMNET++, que permita comparar el desempeño de arquitecturas multi-proveedor en términos de confiabilidad, disponibilidad y costos.
- Validar los resultados obtenidos en el ambiente de pruebas desarrollado, a partir de una comparación con la implementación de algunos casos en el Laboratorio IEC 61850 de la Universidad Nacional.

4. Metodología

La metodología empleada para el desarrollo de este trabajo consiste en modelar cada componente del bus de proceso como un sistema a eventos discretos, utilizando la herramienta OMNET++. Esta herramienta es de libre acceso y permite modelar y simular el comportamiento de sistemas a eventos discretos que interactúan a través de mensajes. Como se verá más adelante, otros autores como (Kumar et al., 2015), (Li et al., 2016) y (Konka, Arthur, García, & Atkinson, 2011) han estudiado el desempeño del bus de proceso mediante simulaciones con otras herramientas similares a OMNET++.

Para alcanzar cada uno de los objetivos específicos de este trabajo, se desarrollaron los siguientes pasos:

- **Estimación de índices de desempeño de dispositivos comerciales**

Se consultó información sobre el desempeño técnico de dispositivos comerciales empleados en el bus de proceso, como *Merging Units*, IEDs de control y protección y *switches* de red. Esto se realizó mediante una búsqueda de información secundaria, teniendo en cuenta fuentes como catálogos y fichas técnicas de fabricantes. La información secundaria fue complementada con consultas directas a los fabricantes, mediante correo electrónico. Toda esta información recopilada fue tabulada y organizada, para definir los parámetros relevantes de cada dispositivo, que serían luego empleados en los modelos de simulación.

- **Caracterización de requerimientos de desempeño del bus de proceso para una subestación típica del SIN colombiano**

Se caracterizó una muestra de subestaciones del SIN colombiano, de niveles de tensión entre los 110kV y los 500kV. Esto se realizó a partir de la búsqueda de diagramas unifilares publicados en documentos de la Unidad de Planeación Minero Energética – UPME. Los diagramas fueron analizados en búsqueda de las configuraciones más comunes y se realizó una entrevista realizada a funcionarios de una empresa de energía local, recibiendo información adicional sobre la disposición de las subestaciones y las posibles arquitecturas de bus de proceso aplicables.

- **Integración de la información en OMNET++**

La información recopilada sobre las subestaciones y las recomendaciones de la literatura consultada fueron útiles para seleccionar una topología de bus de proceso estándar, replicable y redundante, la cual fue implementada en OMNET++. Paralelamente, la información consultada sobre los dispositivos comerciales fue tomada como insumo para definir los modelos de simulación de cada equipo (*Merging Units*, IEDs de control y protección, etc.). Los modelos incluyen parámetros como retardos, número de puertos y cantidad de dispositivos en la red. El diseño del contenido de los mensajes y la dinámica de comunicaciones se basó en las indicaciones de la norma IEC 61850.

- **Validación de resultados**

Se ejecutaron varios ejercicios de simulación de eventos en la red de comunicaciones del bus de proceso, utilizando el ambiente de simulación previamente desarrollado. Algunos de los eventos incluyen la evaluación del retardo de equipos de diferentes fabricantes, la variación del número de equipos en la arquitectura y la pérdida de enlaces de comunicación. Los resultados de estas simulaciones fueron comparados con capturas de tráfico de red de una implementación real de bus de proceso. Esta información fue obtenida en el laboratorio IEC 61850 de la Universidad Nacional y permitió verificar la validez del ambiente de simulación objeto de este trabajo.

5. Marco Teórico

El bus de proceso es el enlace entre los equipos ubicados en patio y los IEDs en el cuarto de control (Mo et al., 2010). El bus de proceso está conformado por los equipos de medición, control y comunicaciones que deben instalarse por cada bahía para efectuar las mediciones de variables como tensión y corriente de bahía y accionar las protecciones en caso requerido.

El bus de proceso está compuesto por los siguientes equipos:

- **IEDs de control:** son los dispositivos encargados de emitir las órdenes de control a los dispositivos de protección, así como de reportar estados de los demás dispositivos y alarmas (Vico, Smith, & Hunt, 2010).
- **IEDs de protección:** se encargan de ejecutar acciones de apertura o cierre de elementos de protección, a partir de la detección de condiciones anómalas en la subestación o de la recepción de mensajes GOOSE de parte del controlador de bahía.
- **Registradores de falla:** son dispositivos que toman registros de variables como tensión y corriente luego de que se presente una falla. Pueden registrar diversos intervalos de tiempo y contar con diferentes frecuencias de muestreo (Pesente, Rolim, Ramos, & Otto, 2016).
- **Switches de red:** son dispositivos que transportan información a diferentes niveles de la red, ejecutando funciones de repetidores o “Hubs”, enlaces entre redes y enrutadores. A diferencia de un Hub tradicional, un *switch* no replica los mensajes entrantes en todos sus puertos de salida, sino que revisa la dirección MAC (Media Access Control) para identificar a cuáles puertos específicos debe replicarse el mensaje. Esto le permite ejecutar acciones de enrutamiento (Mike & Drew, 2002).
- **Merging Units:** son dispositivos encargados de “correlacionar y agregar los datos análogos de las fases individuales o puntos de medición en una subestación, antes de transmitirlos vía Ethernet, para que estén disponibles para otros IEDs” (ABB, 2012).

Determinar la arquitectura de un bus de proceso consiste en definir cuáles clases de equipos se van a utilizar, el número de equipos por barraje y la forma de interconectarlos entre sí. A continuación se describen los principales tipos de arquitecturas de bus de proceso, los parámetros usados para la medición del desempeño de arquitecturas de comunicaciones y las recomendaciones de la norma IEC 61850 sobre la realización del bus de proceso.

5.1. Principales arquitecturas del bus de proceso

Existen unas cinco topologías básicas para realizar el bus de proceso, que son empleadas ampliamente en la literatura:

- Conexiones punto a punto: conecta cada *Merging Units* a un solo relé y un solo controlador. De esta manera, sólo hay una destinación posible para el flujo de datos por fase y no se presentarían retardos por altos volúmenes de paquetes en el mismo canal.
- Configuración en Estrella Dual: introduce un *switch*, como punto común de conexión para el conjunto de elementos relé – *Merging Units* – controlador. Esto permitiría a los IEDs acceder a cada uno de los *switches*, mediante la implementación de un esquema de enrutamiento.
- Configuración en Anillo Simple: se tienen varios grupos *Merging Units* – relé – controlador conectados entre sí por un solo *switch*.
- Configuración de anillo dual: para de reducir tiempos de reconfiguración de rutas en caso de falla en una de las *Merging Units*, se configuran un anillo por cada grupo de equipos relé – *Merging Units* – controlador, pero todos los anillos se conectan al mismo *switch*.
- Configuración en Estrella dual con protocolo PRP: cada grupo relé – *Merging Units* – controlador se conecta a dos *switches*, de manera que un IED cuenta con dos rutas para acceder a cada *Merging Unit*.

La Figura 5 - 1 muestra las arquitecturas de bus de proceso antes mencionadas.

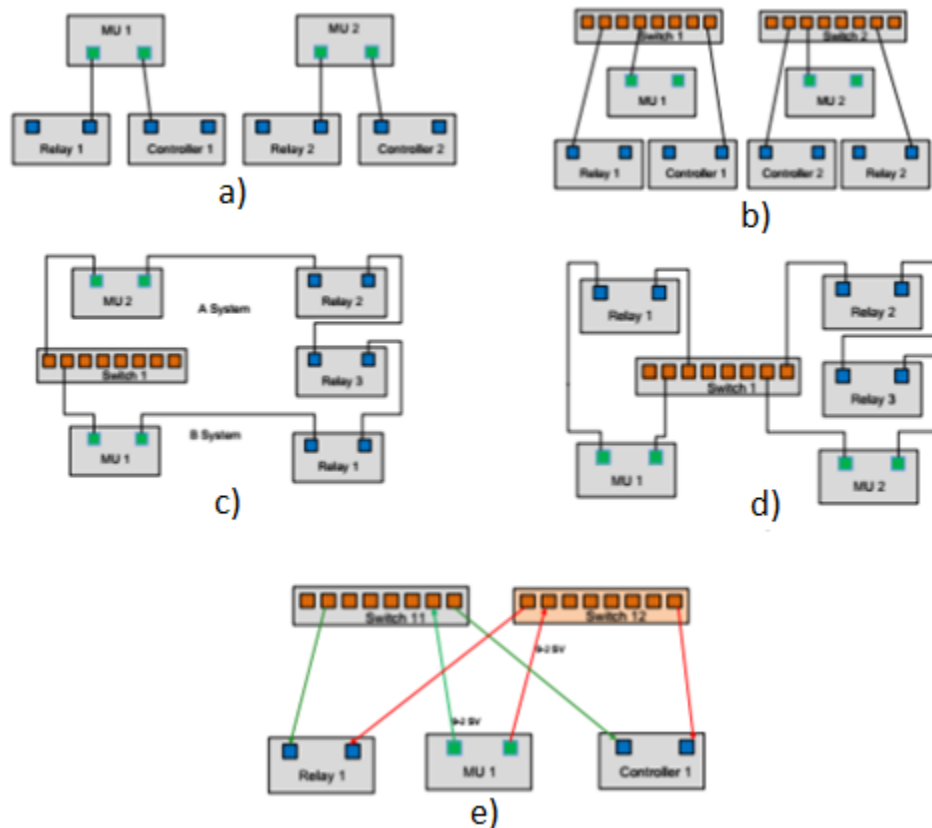


Figura 5 - 1: Posibles arquitecturas del bus de proceso. a) Punto a punto; b) Estrella dual; c) anillo simple; d) anillo dual; e) estrella con PRP (Mo et al., 2010)

Una arquitectura más robusta para el bus de proceso es descrita en (X. Liu et al., 2014). Esta arquitectura se denomina Cobweb y consiste en la interconexión de *switches* de red base con un *switch* de red central, mediante la interconexión de anillos. La Figura 5 - 2 presenta la mencionada topología y la Figura 5 - 3 presenta su aplicación en un bus de proceso. Adicionalmente, la Figura 5 - 4 muestra una topología con comunicación entre bahías, la cual emplea un anillo de mayor jerarquía entre los anillos de cada bahía.

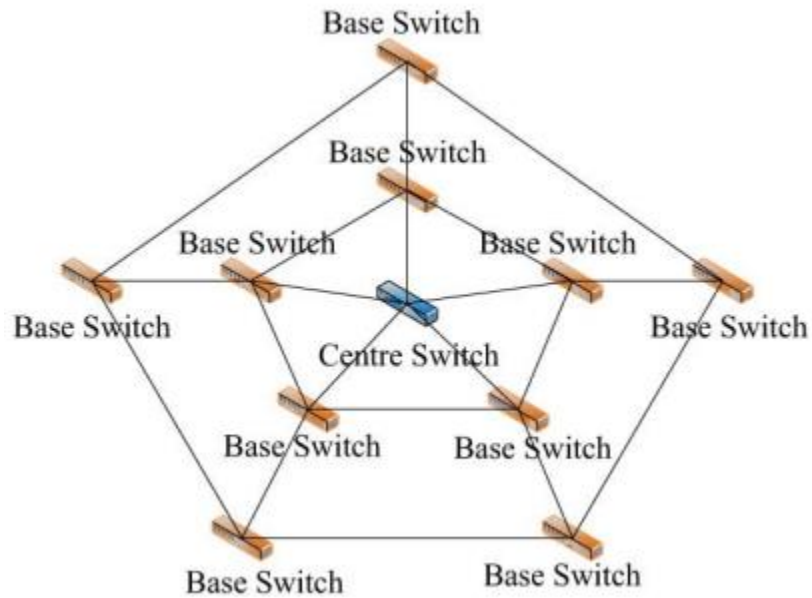


Figura 5 - 2: Topología Cobweb (X. Liu et al., 2014)

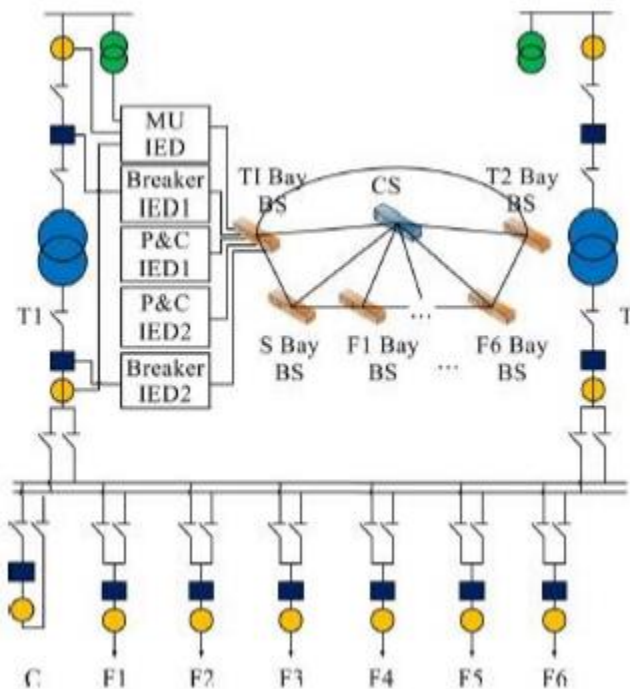


Figura 5 - 3: Topología Cobweb aplicada al bus de proceso (X. Liu et al., 2014)

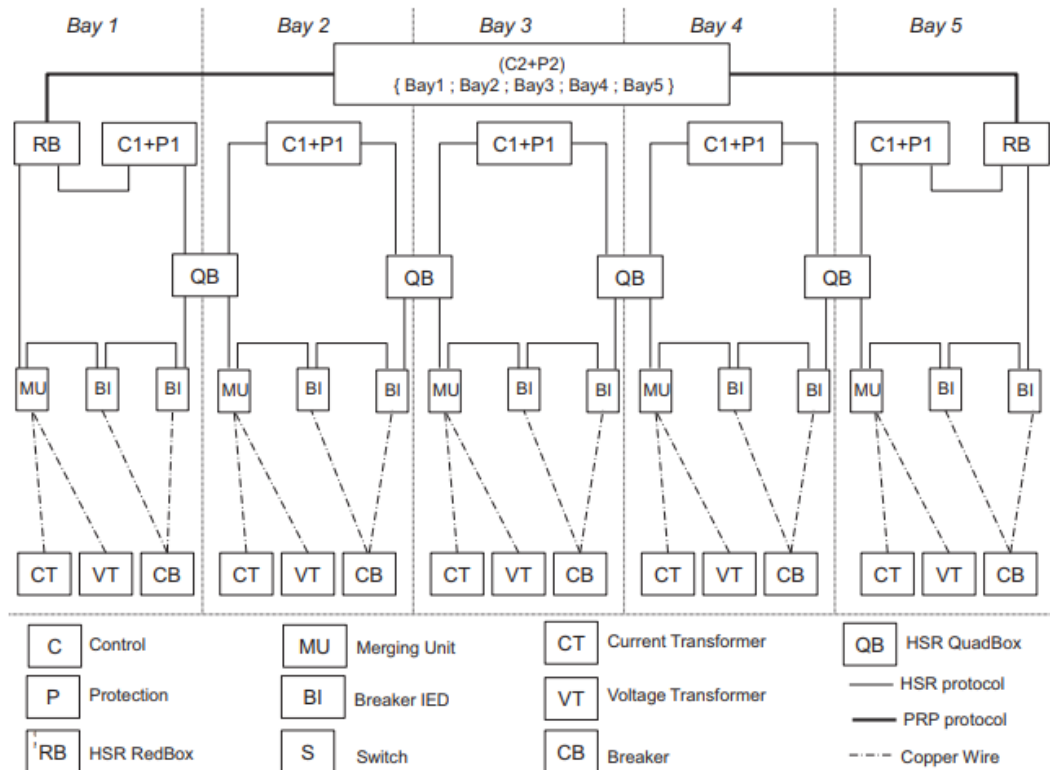


Figura 5 - 4: Arquitectura de comunicación entre controladores de bahía (Tournier & Werner, 2010)

5.2. Evaluación del desempeño de arquitecturas de comunicación

Evaluar el desempeño de un protocolo de comunicaciones consiste en verificar el grado de cumplimiento de los requerimientos de calidad del sistema QoS (Aranda, 2014). Esta calidad del sistema se mide a partir de los siguientes parámetros de desempeño:

- **Pérdida de paquetes:** Es una medida de las fallas en la transmisión de paquetes de información hasta su destino, generalmente causada por congestión en la red. Se define de la siguiente manera:

$$\text{Pérdida de paquetes} = \frac{\# \text{ paquetes recibidos}}{\# \text{ paquetes enviados}} * 100\%$$

Según la norma TiPhone TR 101 329, los estándares de calidad para la pérdida de paquetes se detallan en la Tabla 5 - 1:

Tabla 5 - 1: Requerimientos de QoS para pérdida de paquetes (Sugeng, Istiyanto, Mustofa, & Ashari, 2015)

Pérdida de paquetes Estándar	Categoría	Tasa de pérdida
	Excelente	0%
	Buena	3%
	Media	15%
	Pobre	25%

- **Retardo:** tiempo que tarda la transmisión de un paquete desde un punto a otro de la red. Existe retardo por empaquetamiento, por cola o por propagación. El retardo se puede definir así:

$$\text{Retardo} = \frac{\text{Longitud del paquete (bit)}}{\text{Ancho de banda } (\frac{\text{bit}}{\text{s}})}$$

Según la norma ITU-T G.114 las condiciones de calidad para el retardo se definen en la Tabla 5 - 2:

Tabla 5 - 2: Requerimientos de QoS para el retardo (Sugeng et al., 2015)

Retardo (latencia) Estándar	Categoría	Retardo
	Bueno	0 - 150 ms
	Medio	150 - 400 ms
	Pobre	> 400 ms

- **Jitter:** Es una variación del retardo. Es ocasionado por variaciones en la carga de tráfico y por la cantidad de colisión entre paquetes o congestión de la red. Se calcula con la siguiente fórmula:

$$\text{Jitter} = \frac{\sum \text{variacion en retardo}}{\sum \text{paquetes recibidos}}$$

Según la norma ITU-T G.114 las condiciones de calidad para el jitter se definen en la Tabla 5 - 3.

Tabla 5 - 3: Requerimientos de QoS para el Jitter (Sugeng et al., 2015)

Jitter Estándar	Categoría	Jitter
	Bueno	0 s/d 20 ms
	Medio	20 s/d 50 ms
	Pobre	> 50 ms

- **Throughput:** es el ancho de banda que realmente es empleado para la transmisión de archivos de cierto tamaño, en un tiempo y bajo unas condiciones de red particulares. Se puede definir como la suma de las velocidades de envío de datos a todas las terminales de una red.

$$\text{Throughput} = \frac{\sum \text{datos enviados}}{\text{tiempo de entrega de datos}}$$

De acuerdo con (Sugeng et al., 2015) para Telkom Polytechnic las restricciones de calidad del Throughput se definen en la Tabla 5 - 4:

Tabla 5 - 4: Requerimientos de QoS para el throughput (Sugeng et al., 2015)

Throughput Estándar	Categoría	Throughput/ancho de banda
	Excelente	100%
	Bueno	75%
	Medio	50%
	Pobre	< 25%

Adicionalmente, existen otros índices de desempeño de sistemas en general, que también pueden emplearse para evaluar la confiabilidad de un sistema de comunicaciones. Se trata del tiempo promedio antes de falla (*Mean Time To Failure – MTTF* o *Mean Time Before Failure – MTBF*) y el tiempo promedio de reparación (*Mean Time To Repair - MTTR*). La Tabla 5 - 5 presenta un ejemplo de listado de índices MTTF y MTTR para distintos dispositivos relacionados con el bus de proceso. El costo relativo es adimensional y representa el orden de magnitud del costo de cada dispositivo: por ejemplo, un IED de protección cuesta 100 veces más que los medios de Ethernet y una *Merging Unit* cuesta aproximadamente lo mismo que un controlador de bahía.

Tabla 5 - 5: Índices MTTF y MTTR para algunos componentes del bus de proceso (Liu, Panteli, & Crossley, 2014)

Dispositivo	MTTF (años)	MTTR (horas)	Costo Relativo
IEDs de protección y control de bahía	100	24	10
MU	300	24	4
Switch IED	500	24	3
Switch de Bahía	300	24	4
Switch de estación	250	24	5
Medios de Ethernet	800	24	0.1

5.3. Recomendaciones de la norma IEC 61850 sobre el desempeño del bus de proceso

El apartado IEC 61850-5 describe los tiempos de transferencia en los sistemas de automatización de subestaciones. Se definen dichos tiempos como el espacio de tiempo transcurrido entre la acción de comunicar un valor desde un dispositivo de procesamiento lógico hasta un segundo dispositivo, como parte de una aplicación (Van Rensburg, Dolezilek, & Dearien, 2015). La Figura 5 - 5 muestra los distintos tiempos de acuerdo con la norma, según los dispositivos por los que atraviesa el mensaje. En dicha figura, t_{f1} y t_{f2} representan los tiempos que tardan en activarse las aplicaciones del dispositivo transmisor y receptor, t_a y t_c corresponden a los tiempos de procesamiento de las órdenes en el transmisor y receptor, y t_b representa el tiempo de tránsito por el canal.

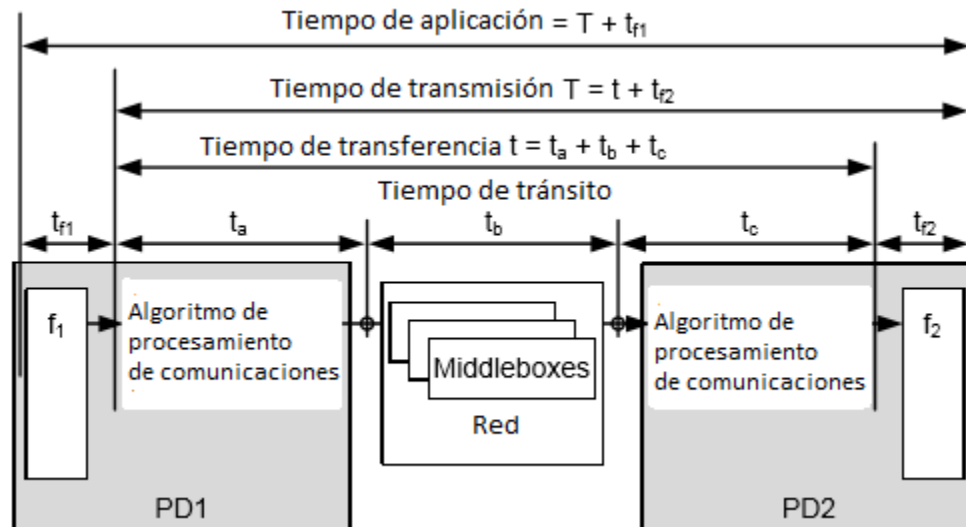


Figura 5 - 5: Tiempos de procesamiento de datos según la norma IEC 61850-5 (Van Rensburg et al., 2015)

Adicionalmente, el reporte técnico IEC/TR 61850-90-4 presenta una guía que brinda mayor claridad sobre los tiempos de procesamiento de los paquetes de comunicación y proporciona el marco de referencia para realizar pruebas de desempeño de retardo en el bus de proceso. Por ejemplo, la Tabla 5 - 6 muestra los requerimientos del reporte técnico para los tiempos de transferencia (IEC, 2013).

Tabla 5 - 6: Requerimientos de tiempos de transferencia según IEC/TR 61850-90-4 (IEC, 2013)

Clase de Tiempo de Transferencia	Tiempo de Transferencia (ms)	Ejemplo de aplicación
TT0	>1000	Archivos, eventos y contenidos de suscripción
TT1	1000	Eventos y alarmas
TT2	500	Comandos de operador
TT3	100	Interacciones automáticas lentas
TT4	20	Interacciones automáticas rápidas
TT5	10	Cambios de estado y liberaciones
TT6	3	Disparos y bloqueos

Los datos presentados en la Tabla 5 - 6 corresponden a diferentes niveles de prioridad para varias aplicaciones en una red bajo IEC 61850. Por ejemplo, el tiempo de transferencia de señales digitales en un esquema de protección corresponden al nivel TT6 (Van Rensburg et al., 2015). Esto debido a la velocidad con que deben actuar las protecciones ante cualquier evento: la IEC 60834 define que el tiempo de transferencia no debe superar los 3 milisegundos y el retardo de recordatorio no debe ser superior a los 18 milisegundos; esto durante el 99,9999% del tiempo de operación de los dispositivos (Chelluri, Dolezilek, Dearien, & Karla, 2014).

Por otro lado, la IEC/TR 61850-90-4 adopta dos protocolos basados en Ethernet para la comunicación en el bus de proceso y bus de estación (IEC, 2013):

- *Parallel Redundancy Protocol* – PRP: se implementan dos redes en estrella redundante y se envía una copia de cada paquete por cada una de las redes.
- *High-availability Seamless Redundancy* – HSR: corresponde a la arquitectura en doble anillo. Cada módulo envía una copia del mensaje en las dos direcciones de circulación del anillo.

6. Estado del arte

A partir del análisis de la literatura reciente sobre el bus de proceso, se pudo determinar que la investigación se ha concentrado principalmente en los métodos para seleccionar una arquitectura de bus de proceso para una subestación en particular y en la evaluación del desempeño de la misma. La mayor parte de las investigaciones se basan en las topologías doble estrella y anillo, y emplean simulaciones para validar los índices de desempeño, como se describe seguidamente.

Con respecto a la realización del bus de proceso, diversas arquitecturas han sido estudiadas por múltiples autores: (Apostolov, Auperrin, Passet, Guenego, & Guilles, 2006) propone una arquitectura similar al anillo dual y asocia a cada nodo lógico una de las funciones que pueden efectuar los diversos dispositivos de una subestación, como por ejemplo la notificación de estado por parte de un interruptor o la señal de apertura por parte de un controlador. Este enfoque también se aprecia en (Mohagheghi, Stoupis, & Wang, 2009). Por otra parte, (Tournier & Werner, 2010) propone las siguientes realizaciones del bus de proceso: Un *switch* de red por bahía (análogo a la configuración estrella); dos *switches* de red por bahía con redundancia (análogo a la configuración estrella con PRP) - Configuración de anillo por bahía; un anillo por bahía con dos IEDs redundantes; y comunicación entre bahías mediante la adición de un anillo de mayor jerarquía entre los anillos de cada bahía.

(Li, Li, & Wen, 2016) también usa la arquitectura estrella simple con un IED mediante un cable de 100Mb/s y un *switch* de red de 32 puertos. A esto se suma (N. Liu et al., 2014), que plantea ocho arquitecturas de bus de proceso, con el fin de evaluar cuál es más confiable. Dichas arquitecturas no son otra cosa que combinaciones de configuraciones estrella – anillo entre el bus de proceso y el bus de estación.

Se puede concluir entonces que las topologías de bus de proceso más empleadas en la literatura son la arquitectura en doble estrella con redundancia PRP y la arquitectura en doble anillo con redundancia HSR.

Métodos para seleccionar la arquitectura de bus de proceso

La investigación reciente se ha concentrado además en los métodos para seleccionar una arquitectura de bus de proceso en función de las subestaciones. Por ejemplo, (Vidiati et al., 2008) define índices de significancia de las subestaciones. Esto sirve para determinar la arquitectura de los buses de proceso y de subestación en función del tipo de datos a comunicar y los requisitos de confiabilidad y seguridad, los cuales serán más o menos estrictos dependiendo de la importancia de la subestación. Los índices propuestos incluyen: Capacidad de la subestación, número de transformadores, tipo de barras, número de alimentadores mayores o iguales a 63kV, ubicación de la subestación en el sistema, significancia de la subestación en el sistema y la significancia de las funciones de operación.

Dependiendo del puntaje que obtenga una subestación al evaluar los anteriores criterios, se escoge una topología de bus de proceso más sencilla o más robusta.

Por otro lado, (Kasztenny et al., 2009) propone que la arquitectura de bus de proceso debe tener en cuenta divisiones por zonas de protección, limitadas por *Current Transformers* – CTs e interruptores. Por lo general, un relé protege una zona, teniendo acceso a los CTs y Breakers vecinos y requiriendo comunicación bidireccional. El artículo analiza también las ventajas de remplazar los esquemas tradicionales de zonas de protección, mediante el uso de MUs, conectadas a pares de IEDs.

En resumen, la literatura muestra que la topología de bus de proceso a seleccionar deberá tener en cuenta el número de CTs y *Voltage Transformers* – VTs de la subestación, lo cual puede influir en el número de *Merging Units*, además de la criticidad de la subestación para robustecer las comunicaciones en el bus.

Evaluación del desempeño del bus de proceso

Una vez definidas las topologías del bus de proceso y la forma de seleccionarlás de acuerdo con la subestación, es importante evaluar qué tan efectiva es una topología en particular. (Tournier & Werner, 2010) define el desempeño de la arquitectura en términos de Confiabilidad y disponibilidad, costo y desempeño. Este trabajo presenta ecuaciones para calcular cada una de estas variables y considera que el desempeño del bus de proceso es una composición entre la carga de la red y peor caso de retraso en transmisión (en caso de falla).

(N. Liu et al., 2014) mide la confiabilidad y disponibilidad del bus de proceso en términos del tiempo medio para falla y tiempo medio para recuperación (*Mean Time To Failure – MTTF* y *Mean Time To Repair – MTTR*). Se consultan los índices MTTF y MTTR para cada uno de los elementos que componen la arquitectura. Estos índices se acumulan para toda la topología, mediante la construcción del Diagrama de Bloques de Confiabilidad – RBD.

Otros trabajos como (X. Liu et al., 2014) también emplean los índices MTTF y la tasa de fallos de cada componente de la red, para comparar la confiabilidad de diferentes arquitecturas de bus y determinar cuál es mejor.

Mayores detalles sobre el cálculo de los índices MTTF y MMTR se presentan en (Jiang & Singh, 2010): los índices se definen en términos de las probabilidades de falla y éxito de un componente particular; estas probabilidades a su vez se definen a partir de las tasas de fallos y las tasas de reparación de cada componente.

Por otro lado, (Mo et al., 2010) mide el desempeño del bus de proceso a partir del tiempo de llegada de los mensajes GOOSE, efectuando una comparación entre los tiempos de llegada de las señales de control mediante un esquema de cableado análogo tradicional y un esquema de bus de proceso.

En la literatura se encuentra que los diferentes autores evalúan los índices de desempeño antes mencionados mediante algún tipo de simulación, en ambientes como OPNET, OMNET++ o NS3. Un ejemplo es (Li et al., 2016), que utiliza OPNET para evaluar el desempeño del control del flujo de datos en el bus de proceso y subestación, a partir de diferentes casos de parámetros como velocidad LAN, tiempo entre llegadas, tamaño de paquetes y volumen de tráfico.

Dicho trabajo asume también una arquitectura tipo estrella entre buses de proceso y se simulan diferentes escenarios: tasa de actualización cliente/servidor de 0.5s, de 1s y de 5s, además de una falla al segundo 1 y al segundo 5. Como resultados de esta simulación se determina la influencia del número de MU conectadas sobre la utilización del ancho de banda y el retardo: Para 18 MU es mayor a 3 ms. Para el bus de subestación, se presentan gráficas del tiempo de retardo y del uso del ancho de banda, para varios números de bahías conectadas.

(Kumar et al., 2015) utiliza OPNET para evaluar el desempeño de un esquema de protección en una subestación de 132kV. Se implementaron 30 CT, 8 VT y una MU por cada grupo de equipos de patio (un CB, 3 CT y 3 VT). Utilizando la herramienta OPNET, se implementa una arquitectura estrella y se evalúan los retardos de los paquetes GOOSE. Como resultados de esta simulación, se presentan gráficas del retardo promedio y los demás parámetros ingresados en OPNET.

(He, He, Zexiang, Xiaohua, & Lin, 2014) simula dos arquitecturas (estrella y anillo) en OPNET, para obtener gráficas de retardos. Como resultado, se evidencia que la arquitectura de anillo presenta un retardo de hasta 62 microsegundos más que la topología estrella.

Por otro lado, algunos trabajos consisten en el desarrollo de herramientas de simulación propias. (Mo et al., 2010) presenta una plataforma para evaluar aplicaciones de bus de proceso. Se evalúan diferentes tipos de tráfico en diferentes arquitecturas de red. La plataforma cuenta con generadores de mensajes GOOSE, SMV, disparidades, un reloj GPS, tráfico de fondo, entre otros. Se probó una arquitectura de anillo simple, para cuatro clases de servicio con siete niveles de prioridad diferentes. Por ejemplo, la publicación de *Sampled Values* la realizan las *Merging Units* a 80 muestras por ciclo, con una trama de 99 bytes, con prioridad 6. Se analizan dos casos: con y sin tráfico de fondo.

(Konka, Arthur, García, & Atkinson, 2011) propone un generador de tráfico *Sampled Value* en otra plataforma de simulación, llamada NS3. Comprende nodos de red que generan paquetes *Sampled Value*, una capa VLAN que permite segmentar el tráfico según IEEE 802.1Q y una subcapa FDEthernet, que imita una tarjeta Ethernet para adicionar direcciones MAC, etiquetas VLAN y secuencias de chequeo.

Para validar el generador de tráfico, se comparan los resultados de la simulación con capturas de Wireshark, obteniendo resultados similares. Sin embargo, este trabajo no considera la creación de paquetes GOOSE o GSSE.

De forma complementaria, (Sidhu et al., 2010) presenta la simulación del tráfico GOOSE en la automatización de subestaciones, empleando el simulador OPNET. Este trabajo cuenta con algoritmos para la programación de paquetes GOOSE y su enrutamiento de acuerdo con determinada prioridad, además de simular el efecto de diferentes enrutamientos en el retardo del envío de paquetes GOOSE. Sin embargo, este trabajo no considera los paquetes *Sampled Values*, ni los paquetes de sincronización PTP.

Finalmente, (Giorgi & Narduzzi, 2007) presenta simulaciones en OMNET++ del procedimiento de sincronización PTP, implementando paquetes de sincronización, de solicitud y respuesta de retardos, además de los módulos maestro y esclavos interactuando en la red. El trabajo muestra la influencia de la carga de tráfico y los errores de la estampa de tiempo en la sincronización del sistema, pero de forma análoga a los dos trabajos anteriormente descritos, sólo se concentra en los paquetes PTP, sin tener en cuenta servicios *Sampled Value* ni GOOSE.

Estos tres trabajos permiten evidenciar que si bien hay investigaciones previas cuyos productos son simulaciones de tráfico de ciertos servicios relacionados con el bus de proceso, son escasos los trabajos que integran los tres servicios *Sampled Values*, GOSSE y PTP. Se concluye pues que buena parte de la investigación se ha centrado en la evaluación de índices de desempeño como el MTTF, MTTR, retardos y congestión en la red del bus de proceso, sin abordar simultáneamente todos los servicios que establece el estándar IEC 61850.

Como se verá más adelante, el presente trabajo es similar a los mencionados anteriormente en el sentido que utiliza herramientas de simulación como OMNET++ y emplea parámetros como tasa de pérdidas y retardos en paquetes para medir el desempeño del bus de proceso. Sin embargo, este trabajo se diferencia de los trabajos presentados previamente porque presenta un enfoque más orientado al desempeño de dispositivos comerciales reales, a partir de la información obtenida en catálogos de fabricantes. De igual manera, el presente trabajo pretende construir un ambiente de simulación que implemente varios servicios de la norma IEC 61850, como *Sampled Values*, GOOSE o PTP, en lugar de concentrarse sólo en el tráfico de un tipo de servicio.

7. Estimación de índices de desempeño para dispositivos comerciales

Uno de los elementos fundamentales de un ambiente de simulación de bus de proceso con orientación a múltiples proveedores, es precisamente la posibilidad de comparar la influencia de dispositivos de diversos fabricantes en el desempeño de una arquitectura concreta. El ambiente de simulación debería implementar diferentes modelos para un mismo dispositivo, en función de las características propias de cada fabricante, de manera que se pueda diferenciar la influencia de cada dispositivo en la red.

Esto requiere de una recopilación de información técnica sobre los dispositivos comerciales, orientada a responder una de las preguntas de investigación de este trabajo: “¿de qué manera influyen factores como el fabricante, el costo y las condiciones de operación, en los índices de desempeño MTTF y MTTR para cada componente del bus de proceso?”

La información técnica sobre dispositivos comerciales se recopiló mediante el siguiente método:

1. Determinación de dispositivos objetivo: teniendo en cuenta el estado del arte, se definieron como dispositivos objetivo de consulta los IEDs de control, los IEDs de protección, los relés, los *switches* y las *Merging Units*, como dispositivos más usados en el bus de proceso.
2. Búsqueda de información secundaria: para cada uno de los dispositivos, se consultaron catálogos, hojas técnicas y manuales de diferentes referencias y de distintos proveedores. En dichos documentos se procuró identificar características que permitieran diferenciar entre fabricantes y referencias, como número de puertos, parámetros MTTR y MTTF, ancho de banda y velocidades de procesamiento.

3. Búsqueda de información primaria: a partir de la búsqueda anterior fue posible establecer una muestra significativa de fabricantes de dispositivos de bus de proceso. Con el objetivo de complementar la información secundaria, se estableció contacto con cada uno de los proveedores de la muestra, mediante correo electrónico o los portales de atención al usuario en las correspondientes páginas web. Los resultados de la correspondencia intercambiada con los fabricantes se pueden consultar en el Anexo A.
4. Tabulación de la información: toda la información recopilada en los dos pasos anteriores fue homologada y tabulada, con el fin de obtener las características de cada dispositivo del bus de proceso en función de cada fabricante.

Los resultados de la aplicación del anterior procedimiento, en cada uno de los dispositivos del bus de proceso, se presentan seguidamente. Como se verá, pese a la hipótesis inicial de que es posible diferenciar los dispositivos mediante parámetros como el MTTR y el MTTF (o MTBF), la investigación mostró que los fabricantes no suelen hacer explícitos dichos parámetros. Además, el MTTR y el MTTF tienen una orientación a muy largo plazo, que no permite evidenciar el desempeño en tiempo real del bus de proceso. Por esta razón, se migró a un análisis de parámetros de desempeño de red, tales como tiempos de procesamiento o retardos.

7.1. IEDs de Control – controladores de bahía

Se identificaron los siguientes fabricantes de IEDs de Control, sin que ninguno especifique tiempos MTTR o MTTF:

- **ABB:** cuenta con las referencias REC670 y PCM 600. REC670 implementa funciones de protección diferencial, protecciones de corriente, tensión y frecuencia, recierres automáticos y control de dispositivos entre una y seis bahías (ABB, 2007). PCM600 corresponde a un IED que puede emplearse para funciones de protección y control, en todos los niveles de tensión (ABB, 2013).
- **Alstom – GE:** la MiCOM C264 tiene capacidad para procesar hasta 5012 entradas digitales, 1024 entradas análogas, 4 CTs y 5 VTs (Alstom, 2015).

- **Siemens:** tiene las referencias de controladores de bahía SIPROTEC 6MD85, SIPROTEC 6MD86, SIPROTEC 6MD63, SIPROTEC 6MD66 Y SICAM BC (Siemens, 2016a, p. 5), (Siemens, 2013a), (Siemens, 2011).
- **Vizimax:** cuenta con una serie de dispositivos que pueden emplearse como controladores de bahía: RightWON Engine Rackmount, RightWON Engine Standalone, RightWON Satellite y RightWON Plus (Vizimax, 2012a), (Vizimax, 2012b).

La información obtenida sobre los IEDs de Control fue organizada y homologada, para poder diferenciar las prestaciones de cada referencia, como se muestra en la Tabla 7 - 1. En esta tabla se observa que podrían ser parámetros útiles los tiempos de procesamiento y el número de entradas y salidas, pero no los tiempos MTTR y MTTF.

Tabla 7 - 1: Parámetros de los IED de control (elaboración propia)

FABRICANTE	REFERENCIA	ENTRADAS CORRIENTE	ENTRADAS TENSIÓN	ENTRADAS BINARIAS	PROTOCOLOS	Tiempo de transmisión	tiempo procesamiento	MEMORIA
ABB	PCM600	ND	ND	ND	DPN3, IEC 60870-5, IEC 61850	ND	ND	ND
ABB	REC670	30 aparatos controlables	30 aparatos controlables	6x32	C37.94, IEC 60870-5, IEC 61850	La transmisión se realiza a 64kbit/s	ND	ND
GE – Alstom	MiCOM C264	1024 entradas análogas 4 transformadores de corriente	1024 entradas análogas 5 transformadores de tensión	5012 entradas digitales 1024 salidas digitales 128 contadores digitales 128 setpoints digitales 16 IEDs por enlace serial 64 indicaciones de posición de taps	UCA2/IEC 61850, IEC 60870-5-104, DNP3, IEC 60870-5-101/103, and MODBUS	tiempo de operación de salidas digitales: 7ms	Programmable Scheme Logic execution time: 5ms	ND
Schneider	MiCOM C264	4 entradas análogas	4 entradas análogas	6 entradas digitales en el Circuit Breaker Unit 4 relés digitales de salida en el CBU 16 entradas digitales en el Digital Input Unit 10 salidas digitales en el Digital Output Unit	T101, DNP3, MODBUS	ND	Tiempo de operación para salidas digitales: 7 ms	ND
SIEMENS	SIPROTEC 6MD85	5 variantes con 4 entradas (transformadores de corriente)	5 variantes con 4 entradas (transformadores de tensión)	de 11 a 75 entradas binarias de 9 a 41 salidas binarias	Control de hasta 20 dispositivos <i>switches</i>	Prioridad 1: 8 - 26 ms Tasa de transmisión: 2048kbit/s Prioridad 2: 9 - 32 ms Prioridad 3: 100 - 200 ms	Pickup time: 3ms dropout time: 4ms Closing time: 4ms Opening time: 2ms, máx 5ms OOT 16ms Operate time: 22ms +	ND

FABRICANTE	REFERENCIA	ENTRADAS CORRIENTE	ENTRADAS TENSIÓN	ENTRADAS BINARIAS	PROTOCOLOS	Tiempo de transmisión	tiempo procesamiento	MEMORIA
							OOT (Cada función tiene su propio Operate time)	
SIEMENS	SIPROTEC 6MD86	6 variantes con 8 entradas (transformado de corriente)	6 variantes con 8 entradas (transformado de tensión)	de 11 a 75 entradas binarias de 9 a 41 salidas binarias	Control de hasta 20 dispositivos <i>switches</i>	Prioridad 1: 8 - 26 ms Tasa de transmisión: 2048kbit/s Prioridad 2: 9 - 32 ms Prioridad 3: 100 - 200 ms	Pickup time: 3ms dropout time: 4ms Closing time: 4ms Opening time: 2ms, máx 5ms OOT 16ms Operate time: 22ms + OOT (Cada función tiene su propio Operate time)	ND
Vizimax	RightWon Plus	Hasta 8 entradas análogas seleccionables		16 ó 32 entradas binarias 16 salidas binarias Form C hasta 12 salidas de relé de alta velocidad	DNP3 Secure V5, IEC 61850, IEC 60870-5-101, -103 and -104, Modbus		Tasa de adquisición: 256 muestras/ciclo	ND
Vizimax	RightWon Satellite	1		ND	DPN3, IEC 61850, Modbus		ND	ND
Vizimax	RightWon Engine Standalone		ND	4 entradas digitales 4 salidas digitales	DPN3, IEC 61850, IEC 60870	ND	ND	RAM: 1GB Flash

FABRICANTE	REFERENCIA	ENTRADAS CORRIENTE	ENTRADAS TENSIÓN	ENTRADAS BINARIAS	PROTOCOLOS	Tiempo de transmisión	tiempo procesamiento	MEMORIA
								sotorage for events: 4GB
Vizimax	RightWon Engine Rackmount		ND	6 puertos LAN 2 puertos USB 4 puertos RS232 8 puertos RS485	RS232, IEC 61850-3, DPN3	La transmisión se realiza a 480Mbps o 12Mbps	ND	RAM: 512MB Compact Flash: 4GB

7.2. IEDs de Protección y relés

Para el caso de estos dispositivos se identificaron tres fabricantes principales:

- **ABB:** la REF615 es una “Protección de alimentación y control”, gestionada desde la herramienta PCM600, la cual es propia del fabricante ABB y permite ejecutar funciones de planeación y configuración de dichos dispositivos (ABB, 2011).
- **Siemens:** cuenta con cinco referencias de dispositivos que pueden configurarse como IEDs de protección en una subestación (Siemens, 2013b, p. 87), (Siemens, 2013c, p. 86), (Siemens, 2016b, p. 85), (Siemens, 2016c, p. 85), (Siemens, 2016d, p. 87):
 - 7SJ82/7SJ85 protección contra sobrecorriente,
 - 7SS85 protección contra baja impedancia de barra
 - 7UT82, 7UT85, 7UT86 y 7UT87 protecciones diferenciales de transformador
 - 7SA82, 7SD82, 7SL82, 7SA84, 7SD84, 7SA86, 7SD86, 7SL86 y 7SJ86 protecciones de distancia y protecciones diferenciales de línea
 - 7SA87, 7SD87, 7SL87 y 7VK87 protecciones de distancia y diferencial de línea.
- **Vizimax:** tiene cuatro familias de relés de protección de diversa naturaleza. La familia SMRB2 comprende relés de enclavamiento que pueden soportar más de 100.000 operaciones y que tardan unos 10 minutos en ser reparados (Vizimax, 1997a, p. 2); la familia SMRS2 comprende relés auxiliares para el control automático para dispositivos de potencia (Vizimax, 1998); la serie SMTS comprende relés de retardo o *Static Time Delay Relays* (Vizimax, 1997b); y la referencia SMTU es un temporizador universal que produce una señal desfasada en el tiempo con respecto a una señal de entrada (Vizimax, 1997c).

La Tabla 7 - 2 presenta un resumen de los parámetros identificados para los IEDs de protección, a partir de los diversos catálogos de fabricantes consultados. Con respecto a los tiempos de procesamiento, se evidencia que para algunas referencias fue posible identificar el tiempo de actuación de los relés y el tiempo de envío de mensajes según su prioridad.

Sin embargo, para equipos como el REF615 no fue posible determinar estos tiempos: el manual especifica diversos valores de retardo, pero son los retardos configurables para cada protección y por lo general se especifican como rangos. Por ejemplo, las funciones de protección 50BF pueden programarse con un retardo de entre 0 y 60.000 ms, con paso de 1ms (ABB, 2011).

Tabla 7 - 2: Parámetros identificados para IEDs de Protección (elaboración propia)

FABRICANTE	REFERENCIA	ENTRADAS CORRIENTE	ENTRADAS TENSIÓN	ENTRADAS BINARIAS	Tiempo de transmisión	tiempo procesamiento
ABB	REF615	8 entradas análogas X130	ND	52 entradas binarias	3ms	ND
SIEMENS	SIPROTEC 7SL87	Trans. Protec: 3/5A; 3/1A Trans. Medida: 1/5A; 1/1A	5 módulos IO de 200V	ND	Prioridad 1: 8 - 26 ms Prioridad 2: 9 - 32 ms Prioridad 3: 100 - 200 ms	Tiempo de cierre: 0.01 - 0.6 s
SIEMENS	SIPROTEC 7SJ82	Trans. Protec: 1/5A; 1/1A Trans. Medida: 1/5A; 1/1A	5 módulos IO de 200V 8 (el IO212 tiene 8 canales de entrada que pueden usar como entradas de corriente o tensión diferencial)	ND	Prioridad 1: 8 - 26 ms Prioridad 2: 9 - 32 ms Prioridad 3: 100 - 200 ms	Tiempo de cierre: 0.2ms Tiempo de apertura: 6ms máximo 9ms OOT: 16ms
SIEMENS	SIPROTEC 7SA86	ND	de 4 a 8 transformadores de tensión/corriente	de 5 a 31 entradas binarias de 8 a 46 salidas binarias	ND	Tiempo de cierre: 0.2ms Tiempo de apertura: 6ms máximo 9ms OOT: 16ms Tripping times: 23ms+OOT Dropout times: 34 to 49ms + OOT
SIEMENS	SIPROTEC 7UT82	ND	1 variante con 8 transformadores de corriente	7 entradas binarias 7 salidas binarias	Cada función de protección tiene sus propios Dropout time e Idiff	Tiempo de cierre: 0.2ms Tiempo de apertura: 6ms máximo 9ms OOT: 16ms Idiff 20ms+OOT Dropout time: 26m
SIEMENS	SIPROTEC 7SS85	20 puntos de medida trifásicos (transformadores de corriente)	4 puntos de medida trifásicos (transformadores de tensión)	26 bahías, 4 zonas de buses	Prioridad 1: 8 - 20 ms Prioridad 2: 9 - 25 ms Prioridad 3: 100 - 150 ms	OOT: 10ms Fast relay: closing time: 4ms opening time: 2ms, max 5ms Pickup time, internal star: 1ms Pickuptime external start: 5ms

FABRICANTE	REFERENCIA	ENTRADAS CORRIENTE	ENTRADAS TENSIÓN	ENTRADAS BINARIAS	Tiempo de transmisión	tiempo procesamiento
						Dropout time: 15ms Para Zero crossing supervision: pickup time 27ms
Vizimax	SMTS NTTS0323	ND	ND	ND	ND	Operating delay: de 6 a 25 ms
Vizimax	SRB2 NTRB0326	ND	ND	ND	ND	Operation time typical: 14ms Operation time max: 50ms
Vizimax	SMRS2 NTE0322	ND	ND	ND	ND	Pickuptime: typ 10ms, mín 7ms, máx 13ms Dropout time: typ 15ms, mín 8 ms, máx 24ms
Vizimax	SMTU NTTU0324	ND	ND	8 salidas digitales INV por función		4ms para DC outputs

7.3. Switches

Se identificaron tres fabricantes principales:

- **Alstom – GE:** Este fabricante produce la serie de *Switches* Ethernet llamada MiCOM H, que soportan HSR y PRP, con velocidades de transmisión de 10 o 100Mbps, como se aprecia en la Tabla 7 - 3 (Alstom, 2012a).

Tabla 7 - 3: Datos técnicos de diferentes referencias MiCOM (Alstom, 2012a)

Type	Description
MiCOM H15x PCI board (Self Healing Ring)	
H152	10/100 Mbps, 4 TX and 2 FX (ST, multi mode, redundant), SNMP managed
H154	10/100 Mbps, 4 TX and 2 FX (ST, single mode, redundant), SNMP managed
MiCOM H16x PCI board (Dual Homing Star)	
H162	10/100 Mbps, 4 TX and 2 FX (ST, multi mode, redundant), SNMP managed
H164	10/100 Mbps, 4 TX and 2 FX (ST, single mode, redundant), SNMP managed
Connection kit for MiCOM H1xx boards	
H1xx-kit	Rear PC plate with 3 additional RJ45 connectors + connection cables
MiCOM H34x DIN rail mountable switch for simple ethernet star	
H342	10/100 Mbps, 6 TX and 2 FX (ST, multi mode), SNMP managed, dual power supply
H344	10/100 Mbps, 6 TX and 2 FX (SC, single mode), SNMP managed, dual power supply
MiCOM H35x DIN rail mountable switch (Self Healing Ring)	
H352	10/100 Mbps, 6 TX and 2 FX (ST, multi mode, redundant), SNMP managed, dual power supply
H354	10/100 Mbps, 6 TX and 2 FX (SC, single mode, redundant), SNMP managed, dual power supply
H356	10/100 Mbps, 2 TX, 2 FX (LC, multi mode, redundant) and 4 FX (LC, multi mode), SNMP managed, dual power supply
H358	10/100 Mbps, 2 TX, 2 FX (LC, single mode, redundant) and 4 FX (LC, multi mode), SNMP managed, dual power supply
MiCOM H36x DIN rail mountable switch (Dual Homing Star)	
H362	10/100 Mbps, 6 TX and 2FX (ST, multi mode, redundant), SNMP managed, dual power supply
H364	10/100 Mbps, 6 TX and 2FX (SC, single mode, redundant), SNMP managed, dual power supply

- **Siemens:** tiene las referencias: M2100, RSG2100 y RS940G (Siemens, 2013d) (Siemens, 2015a) (Siemens, 2013e).
- **Schneider:** tiene tres familias de *switches* Ethernet: MiCOM H16x, MiCOM H18x y MiCOM H36x (Schneider, 2012a), (Schneider, 2014, p. 18), (Schneider, 2012b, p. 36).

La Tabla 7 - 4 presenta un resumen de los principales parámetros de red identificados para los *switches*, a partir de la información contenida en catálogos y manuales técnicos de los fabricantes.

Tabla 7 - 4: Parámetros de desempeño para *switches* (elaboración propia)

FABRICANTE	REFERENCIA	PUERTOS	TASA DE TRANSFERENCIA	Tiempo de transmisión	tiempo procesamiento
SIEMENS	RUGGEDCOM M2100	19	10Mbit/s, 100Mbit/s, 1000Mbit/s	Tiempo de detección de arco 23ms Delay setting with inst. O/C Prot. 70ms	Forward delay: tiempo que el <i>switch</i> tarda en aprender las MACs en un puerto antes de iniciar el tráfico: 4 a 30 s típico 15s
SIEMENS	RUGGEDCOM RSG2100	19	10Mbit/s, 100Mbit/s, 1000Mbit/s	Tiempo de detección de arco 23ms Delay setting with inst. O/C Prot. 70ms	Período de latencia: 4us, según la página https://support.industry.siemens.com/cs/pd/463149?pdtd=td&lc=en-WW
SIEMENS	RUGGEDCOM RS940G	8	10Mbit/s, 100Mbit/s, 1000Mbit/s	ND	Switch latency period: 4us, según la página https://support.industry.siemens.com/cs/pd/484407?pdtd=td&pnid=22006&lc=en-WW
					Forward delay: 4 a 30 segundos, típico 15
					Reinit delay: 1 a 10s, igual que los otros <i>switches</i>
Alstom	MiCOM H152	4	10Mbps, 100Mbps	ND	ND
Alstom	MiCOM H162	4	10Mbps, 100Mbps	ND	ND
Alstom	MiCOM H342	6	10Mbps, 100Mbps	ND	ND
Alstom	MiCOM H364	6	10Mbps, 100Mbps	ND	ND
Schneider	MiCOM H16X	4	10Mbps, 100Mbps	ND	ND
Schneider	MiCOM H18X	2	100Mbps	ND	ND
Schneider	MiCOM H36X	6	10Mbps, 100Mbps	ND	Latencia: 4us

7.4. *Merging Units*

Se identificaron los siguientes fabricantes:

- **ABB:** cuenta con las referencias CP-MU, SAM600-CT-DA-B, SAM600-CT-DA-B, SAM600-VT-DA-H y SAM600-VT-DA-P2 (ABB, 2012), (ABB, 2016).
- **Alstom - GE:** tiene seis referencias de MU: COSI-MU, M87X-TM, MU Agile AMU, MU AGILE XMU 800, MU AGILE 820, REASON MU320 (GE, 2014), (GE, 2017), (Alstom, 2012b), (Alstom, 2013a) y (Alstom, 2013b).
- **Siemens:** produce los dispositivos 6MU805 Y 7SC805 (Siemens, 2015b) y (Siemens, 2014).
- **Vizimax:** MGU010000 cuenta con una entrada análoga para tensión y otra para corriente, las cuales permiten conectar las tres fases y su respectivo neutro (Vizimax, 2017).

La Tabla 7 - 5 muestra el resumen de los datos obtenidos sobre las *Merging Units* en catálogos de fabricantes y mediante solicitudes por correo electrónico, evidenciando diferencias en parámetros como el número de entradas análogas CT y VT, la presencia de puerto para sincronización IRIG-B, el tiempo de procesamiento y la tasa de error de bits – BER.

Tabla 7 - 5: Parámetros de red para las *Merging Units* (elaboración propia)

FABRICANTE	REFERENCIA	CT INPUTS análogas	VT INPUTS análogas	Entradas digitales	Hardware response digital inputs	Puertos Ethernet	Bitrate	Puertos de entrada	Puertos de salida	IRIG-B	Tiempo de procesamiento	Ancho de banda Tasa de transmisión	Bit Error Rate
Alstom – GE	MU-Agile XMU800	0	0	13	ND	13	ND	6 (PCBUS)	7: 4LC y 3RJ45	1 (patente)	ND	ND	ND
Alstom – GE	MU-Agile XMU820	4 (1A a 5A)	6 (100 a 600 VAC)	13	ND	13	ND	6	7	ND	ND	ND	ND
Alstom – GE	Reason MU 320	8 (5A, ancho de banda 3kHz)	8 (115V, ancho de banda 3kHz)	28	ND	2 inout	100Mbps	12	16	1	4,2 ms retardo del grupo durante adquisición de datos	100Mbps	ND
Alstom	COSI-MU	1CT	1VT	ND	ND	ND	ND	ND	ND	ND	ND	ND	ND
Alstom	MU Agile AMU	4	4	ND	ND	ND	ND	ND	ND	ND	ND	ND	ND
Vizimax	AMU Vizimax	4 (Ia, Ib, Ic e In. 1A a 5A)	4 (Va, Vb, Vc, Vn. 69 a 220 VAC)	10 (6+4)	<0.1ms a tensión nominal <1ms a 80% de tensión nominal	3	10/100M bps	10 (6 con un común y 4 con su propio común)	3	1	ND	ND	ND
ABB	SAM600-CT	4 (Ia, Ib, Ic e In. 1A a 5A)	0	4 inout	ND	4 inout (2 100BASE-TX y 2 SFP)	100Mbps	4	4	0 (requiere SAM- TS)	2ms entre toma de muestra y envío de SV	100Mbps	<10e-6 (Para PMU RES670)
ABB	SAM600-VT	0	4 (Va, Vb, Vc, Vn. 69 a 125 VAC)	4 inout	ND	4 inout (2 100BASE-TX y 2 SFP)	100Mbps	4	4	0 (requiere SAM- TS)	2ms entre toma de muestra y envío de SV	100Mbps	<10e-6 (Para PMU RES670)
ABB	CP-MU	9	ND	ND	ND	ND	ND	ND	ND	ND	3ms	ND	ND

FABRICANTE	REFERENCIA	CT INPUTS análogas	VT INPUTS análogas	Entradas digitales	Hardware response digital inputs	Puertos Ethernet	Bitrate	Puertos de entrada	Puertos de salida	IRIG-B	Tiempo de procesamiento	Ancho de banda Tasa de transmisión	Bit Error Rate
Siemens	SIPROTEC 6MU805	4 (Ia, Ib, Ic e In. 1A a 5A)	4 (Va, Vb, Vc, Vn. 0 a 220 VAC)	ND	5 a 7 us	2 inout	100Mbps	12 binary para indicar estados	8 binary para indicar estados	1	10000 ticks máx. Operaciones entre 1 y 83 ticks	100Mbps	ND
Siemens	SIPROTEC 7SC805	4 (Ia, Ib, Ic e In. 1A a 5A)	4 (Va, Vb, Vc, Vn. 0 a 220 VAC)	ND	5 a 7 us	2 inout	100Mbps	12 binary para indicar estados	8 binary para indicar estados	1	10000 ticks máx. Operaciones entre 1 y 83 ticks	100Mbps	ND

Con toda la información recopilada se puede concluir que muy pocos fabricantes especifican parámetros como el MTTF y MTTR. Por otro lado, estos parámetros suelen tener valores del orden de decenas de años, lo cual hace que su uso sea impráctico en simulaciones de desempeño de una topología de red. Por esta razón, el ambiente de simulación desarrollado se concentró en los parámetros que afectan el desempeño de la red de telecomunicaciones, como tiempos de procesamiento, retardos en los canales, ancho de banda, número de puertos y *Bit Error Rate* – BER.

Por otra parte, la información presentada en las tablas anteriores permite evidenciar que es difícil encontrar información unificada y homologada sobre los parámetros de desempeño de los dispositivos: algunos fabricantes hacen muy explícita y detallada la información sobre los tiempos de retardo, mientras otros no hacen mención alguna a ella; en algunas fichas técnicas es difícil encontrar información sobre el número de puertos del dispositivo, mientras que para otros fabricantes esta es la única información suministrada.

Como se verá en el capítulo 9, esta falta de homologación de la información dificulta la obtención de modelos confiables de los dispositivos comerciales, por lo que se recomienda en lo posible validar los modelos mediante pruebas en laboratorios, como el Laboratorio IEC 61850 de la Universidad Nacional.

8. Caracterización de requerimientos de desempeño del bus de proceso para una subestación típica del SIN colombiano

8.1. Características de las subestaciones del SIN

Generalidades de las subestaciones eléctricas

Una subestación eléctrica es un conjunto de infraestructura y equipos que permiten efectuar la interconexión de diferentes tramos de un sistema de potencia, mediante el cambio de los niveles de tensión y corriente o mediante la adecuada convergencia de diferentes líneas de transmisión (Hidalgo Quesada, 2008). Pueden distinguirse los siguientes tipos de subestaciones:

- Subestaciones de generación: elevan la tensión desde los niveles de generación (típicamente con valores entre los 5kV y los 25kV) hasta los niveles de transmisión, desde 69kV en adelante.
- Subestaciones receptoras primarias: disminuyen la tensión desde las líneas de transmisión hasta niveles adecuados para la distribución. Por lo general, operan con 134kV en el lado de alta y con 69kV o 34.5kV en el lado de baja tensión.
- Subestaciones receptoras secundarias: son equivalentes a las estaciones de distribución que toman la tensión en niveles de subtransmisión y la llevan a niveles de distribución de 34.5kV

En cuanto a las configuraciones de las subestaciones de potencia son comunes las siguientes (United States Department of Agriculture, 2001), (Mayorga & Sarmiento, 2010):

- **Subestaciones de barra sencilla:** todos los elementos de operación y transformación están conectados a un solo juego de barras. Siendo la configuración más sencilla, es también la más barata, compacta y fácil de mantener.
- **Barra sencilla particionada:** cuenta con un seccionador en la barra única, de manera que se puede aislar parte de la subestación en caso de que una de las bahías presente una falla, sin afectar la operación de todas las demás bahías.
- **Doble barra (europea):** tiene dos barras que pueden acoplarse con un disyuntor y sus seccionadores. Cada bahía se conecta mediante disyuntores y seccionadores a cada una de las barras, obteniendo así una redundancia.
- **Configuración en anillo:** esta configuración es muy empleada en subestaciones de distribución. Las bahías se conectan entre sí y a las barras mediante disyuntores y seccionadores, formando una especie de anillo.
- **Configuración de interruptor y medio:** este esquema es muy utilizado en subestaciones de alta potencia. Consiste en dos barras, cada una normalmente energizada. Entre las dos barras hay tres interruptores (o disyuntores). En el medio de cada par de interruptores, se conecta un circuito, de manera que se tienen siempre tres interruptores y dos circuitos entre las barras.
- **Doble barra con doble interruptor:** consiste en dos barras normalmente energizadas, entre las cuales se ubican pares de interruptores. Entre cada par de interruptores, se coloca un único circuito o bahía, de manera que cada circuito cuenta con dos interruptores independientes.

Como se verá a continuación, la mayor parte de las subestaciones del Sistema de Transmisión Nacional presentan la configuración de interruptor y medio o doble barra más acople de transferencia.

Subestaciones en el Sistema de Transmisión Nacional

El Sistema de Transmisión Nacional colombiano contaba en 2013 con unos 10.267 km en líneas de transmisión de 100-115 kV, 15.5 km en líneas de 138 kV, 11.680 km de líneas de 220-230 kV y 2.437 km en líneas de 500kV, para un total de casi 24.400 km de líneas de transmisión (Rodríguez, 2013).

El diagrama unifilar del Sistema de Transmisión Nacional se puede apreciar en la Figura 8 - 1. Las subestaciones de 220 kV aparecen en color verde y las de 500 kV en rosa.

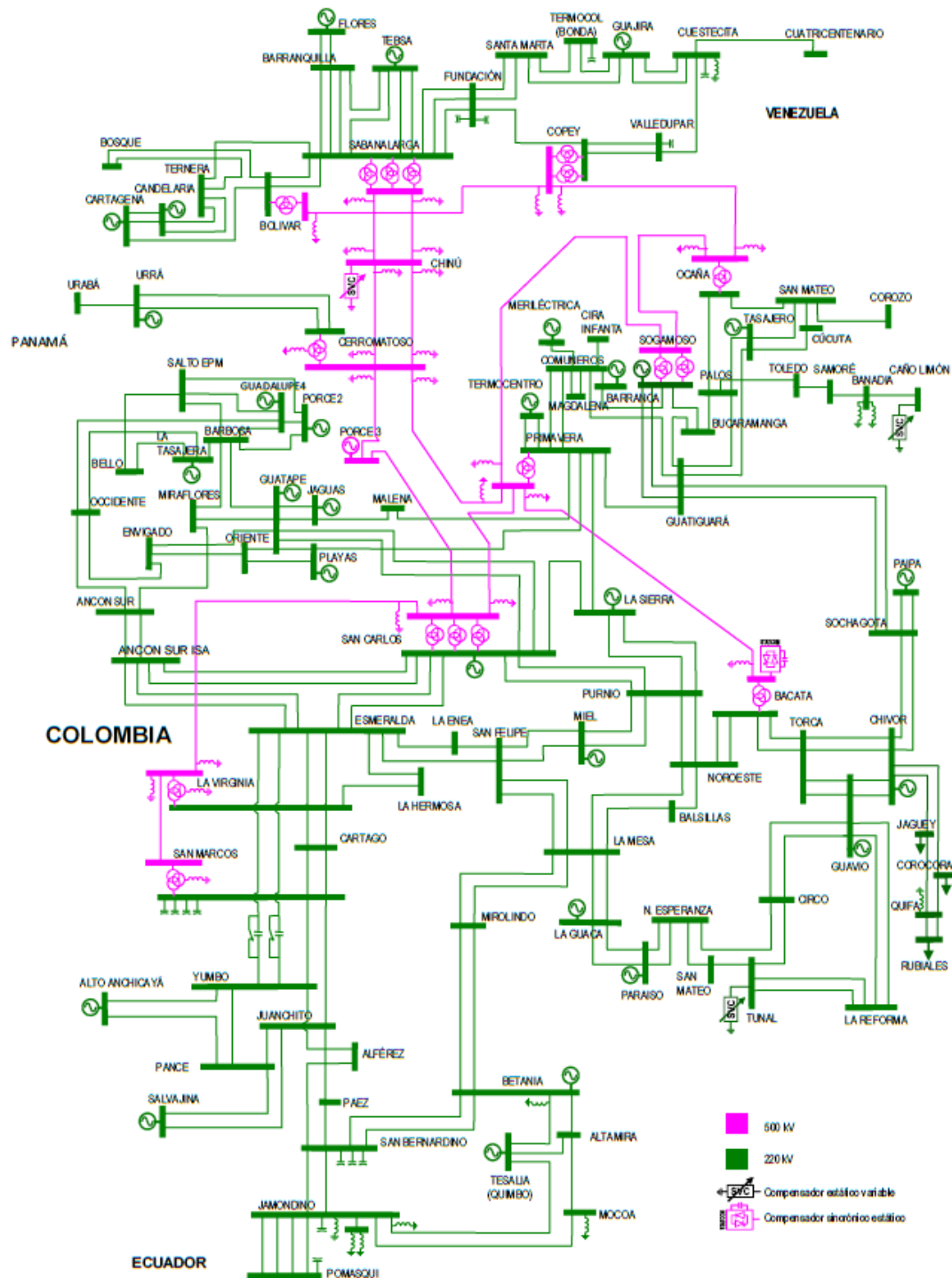


Figura 8 - 1: Diagrama unifilar del Sistema de Transmisión Nacional. Fuente: (UPME, 2016)

Las subestaciones de la Figura 8 - 1 son susceptibles de automatizarse siguiendo los lineamientos del estándar IEC 61850. Para ello, debe tenerse en cuenta la configuración de cada subestación particular, identificando las bahías, los equipos y las funciones de protección necesarias, de manera que se puedan tener insumos para el diseño de los buses de proceso y estación.

Con el objetivo de identificar los requerimientos del bus de proceso para dichas subestaciones, se tomó una muestra de subestaciones de 220kV y subestaciones de 500kV. El Sistema de Transmisión Regional cuenta con 17 subestaciones de 500kV y unas 96 subestaciones de 220-230kV (UPME, 2015).

Se consultaron los diagramas unifilares de algunas de estas subestaciones, para identificar las configuraciones más comunes y el número de equipos. Estos diagramas fueron obtenidos de diversas convocatorias de la UPME para la expansión de las subestaciones. La Figura 8 - 2 muestra un ejemplo de unifilar para la subestación Bolívar 220kV, en configuración doble barra más acople de transferencia; la Figura 8 - 3 muestra un ejemplo de unifilar en configuración interruptor y medio para la subestación Chinú 500kV. El listado completo de diagramas se encuentra en el Anexo B.

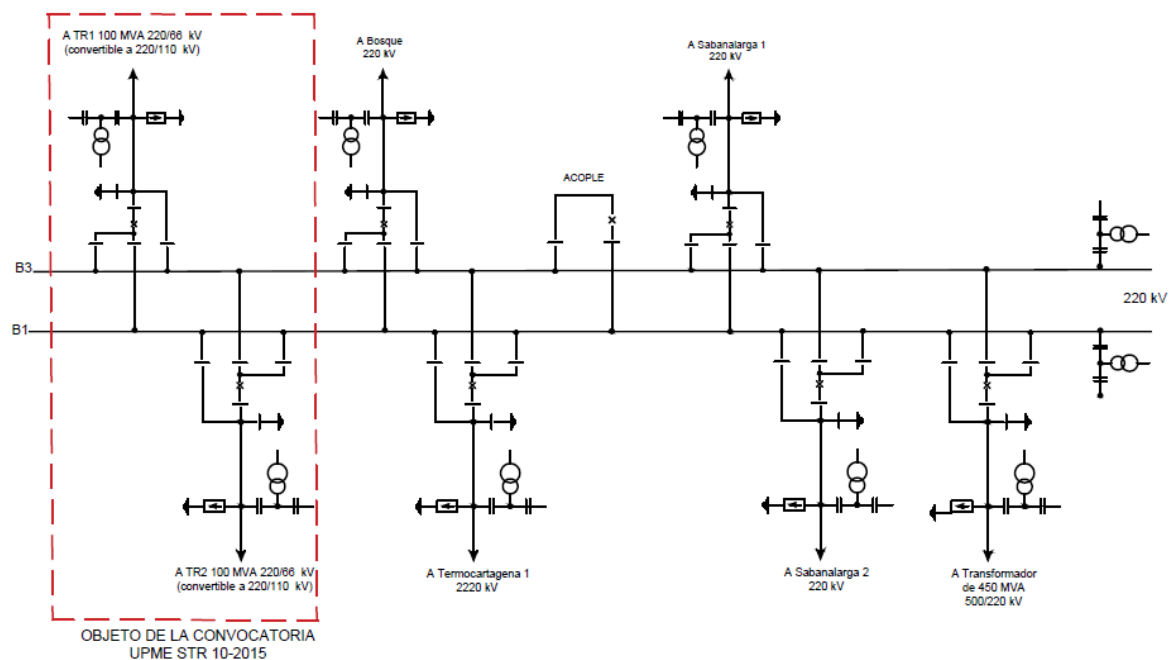


Figura 8 - 2: Diagrama unifilar de la subestación Bolívar 220kV (UPME, 2010)

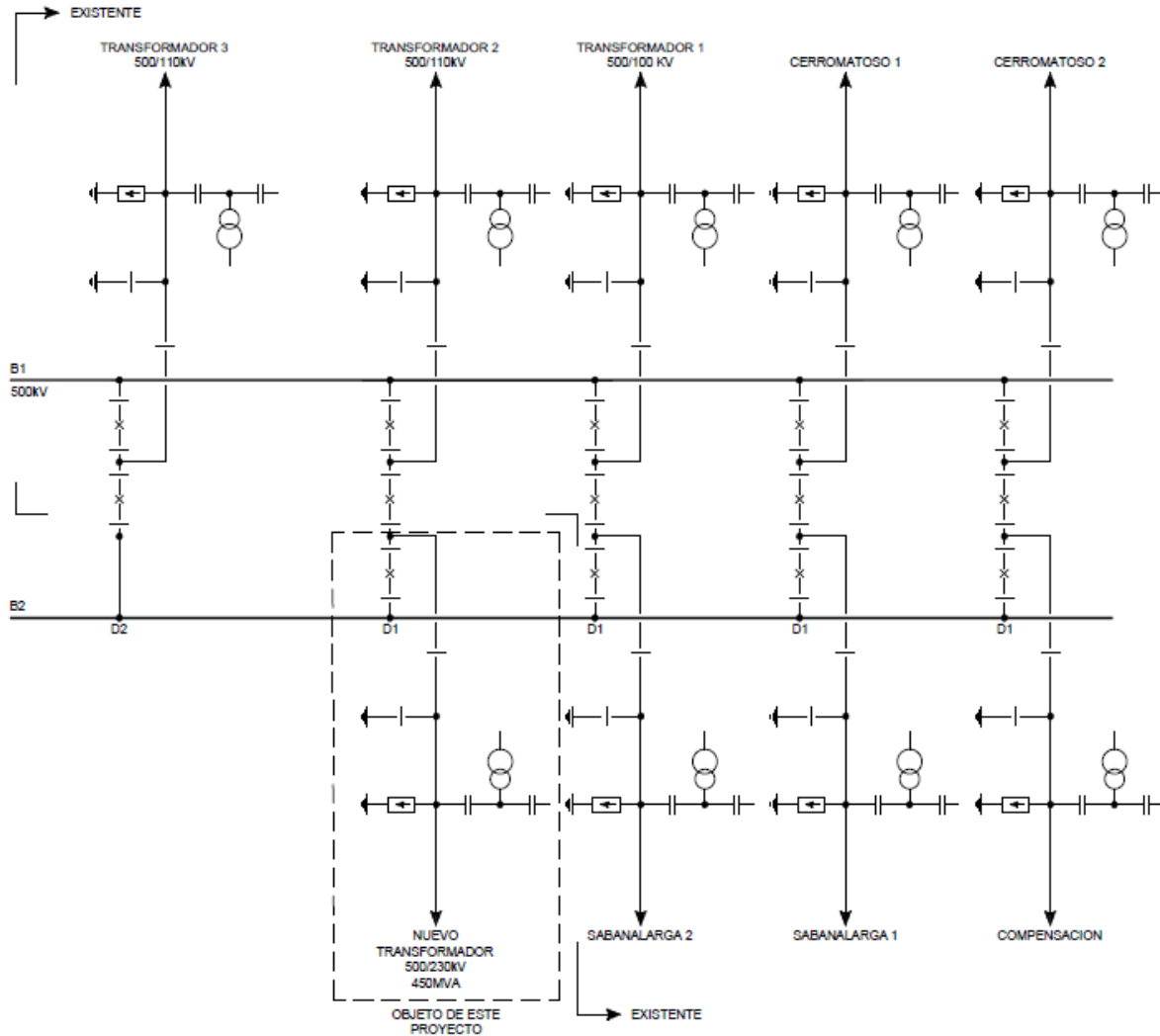


Figura 8 - 3: Diagrama unifilar de la subestación Chinú 500kV (UPME, 2013)

A partir de la búsqueda realizada, fue posible obtener los diagramas unifilares de unas 10 subestaciones de 500kV y de unas 20 subestaciones con niveles de tensión entre los 110kV y los 230kV. Esta muestra representa el 59% de las subestaciones del 500kV y el 21% de las subestaciones de 230kV.

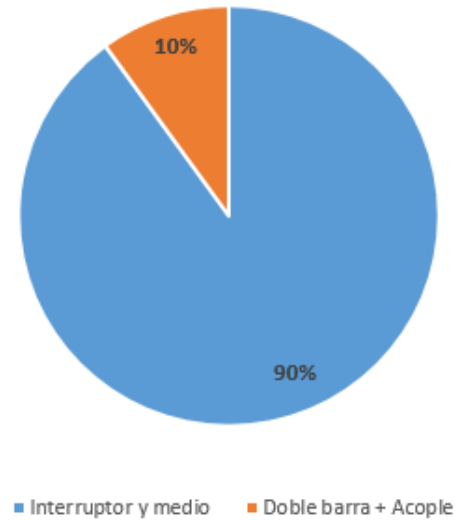
La Tabla 8 - 1 muestra los resultados obtenidos del análisis de los unifilares de 500kV: el 90% de la muestra corresponde a una configuración de interruptor y medio (véase Figura 8 - 4). El número promedio de circuitos por subestación es 8.

Por su parte, la Tabla 8 - 2 muestra los resultados del análisis de los unifilares de subestaciones de 220-230kV, aunque también se incluyen algunas como Termoflores, cuyo nivel de tensión es 110kV. En este caso, las configuraciones son un poco más diversas que para el caso de 500kV: 55% de las subestaciones tienen la configuración de doble barra más acople de transferencia y el 35% presentan configuración de interruptor y medio, como se puede apreciar en la Figura 8 - 5.

Tabla 8 - 1: Configuraciones y circuitos en subestaciones de 500kV (elaboración propia)

Subestación	Configuración	Circuitos
Alfárez	Interruptor y medio	4
Cerromatoso	Interruptor y medio	15
Chinú	Interruptor y medio	9
Copey	Doble barra + acople	5
Ituango	Interruptor y medio	16
La Virginia	Interruptor y medio	7
Nueva Esperanza	Interruptor y medio	6
Porce III	Interruptor y medio	6
San Marcos	Interruptor y medio	3
Sogamoso	Interruptor y medio	10

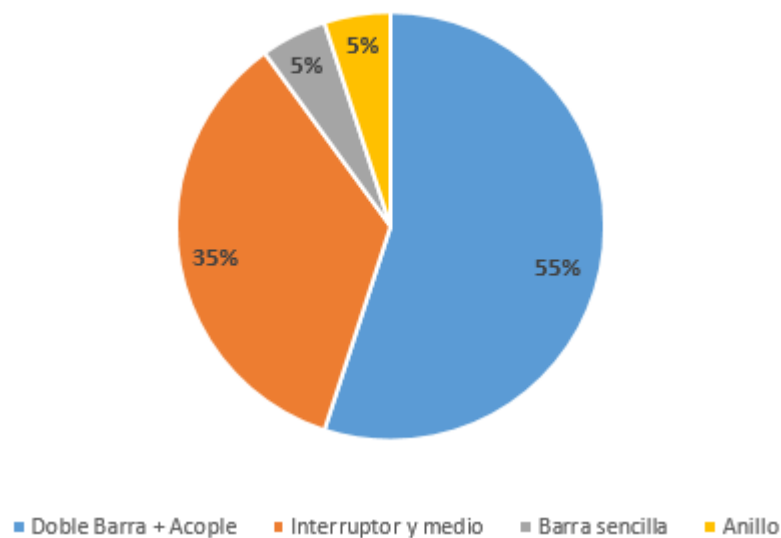
Configuración de las subestaciones 500kV

**Figura 8 - 4:** Configuraciones de las subestaciones de 500kV (elaboración propia)**Tabla 8 - 2:** Configuraciones y circuitos en subestaciones 110-220-230kV (elaboración propia)

Subestación	Configuración	Circuitos
Altamira	Doble barra + acople	6
Ancón Sur	Barra Sencilla	8
Bolívar	Doble barra + acople	9
Bosque	Interruptor y medio	6
Caracolí	Interruptor y medio	6
Cartagena	Interruptor y medio	9
Copey	Doble barra + acople	6
Cuestecitas ISA	Doble barra + acople	5
Cuestecitas Electricaribe	Doble barra + acople	10
Cuestecitas Transelca	Doble barra + acople	5
La Enea	Doble barra + acople	5
La Loma	Doble barra + acople	4
Nueva Barranquilla	Doble barra + acople	8

Subestación	Configuración	Circuitos
San Antonio	Interruptor y medio	4
Tasajero	Interruptor y medio	8
TEBSA	Interruptor y medio	12
Termocol	Interruptor y medio	10
Termoflores	Doble barra + acople	10
Urabá	Doble barra + acople	3
Valledupar	Anillo	9

Configuración de las subestaciones 110-220-230kV

**Figura 8 - 5:** Configuraciones de las subestaciones 110-220-230kV (elaboración propia)

Se puede concluir entonces que la configuración más común para las subestaciones de 500kV es la de interruptor y medio, con un promedio de 8 circuitos por subestación. Para las subestaciones de 230kV la configuración preferida es doble barra más seccionador de acople con un promedio de 7 circuitos por subestación.

Con esta información, se procedió a determinar las arquitecturas de bus de proceso recomendables para estas subestaciones. Para ello, se revisaron especificaciones de la UPME sobre el diseño de las subestaciones y las recomendaciones del estándar IEC 61850, como se describe a continuación.

8.2. Determinación de topologías de bus de proceso recomendables para las subestaciones del SIN

Las especificaciones de las subestaciones fueron determinadas a partir de las especificaciones suministradas por la UPME en diferentes convocatorias para expansión o mantenimiento de subestaciones del SIN.

Especificaciones UPME para subestaciones de 220kV

El siguiente análisis se basa en las especificaciones de la subestación Termocol 220kV, según (UPME, 2012).

Los equipos empleados en la subestación deben cumplir con norma IEC, ANSI, ITU-T y CISPR. Los transformadores de tensión deben cumplir con IEC 60186, IEC 60358 e IEC 60044-4. Más detalladamente, la UPME exige lo siguiente:

- Los transformadores de corriente deben cumplir con la norma IEC 60044.
- Los equipos de protección deben cumplir con la norma IEC 60255 e IEC 60870.
- Los registradores de falla deben cumplir la norma IEEE C37.111.
- Equipos de protección: En cuanto al esquema de protecciones, éste debe contemplar una protección principal y otra secundaria, de diferente principio de operación, además de un registrador de fallas.
- Sincronización: Adicionalmente, la UPME exige la sincronización satelital de todos los equipos de control, protección y registro de fallas, mediante un reloj GPS.
- Los *switches* de red deben cumplir con la norma IEC 61850-3
- Equipos de medición: deben tomar señales de tensión, corriente, potencia activa, potencia reactiva, factor de potencia y frecuencia. Deben cumplir las disposiciones de la Resolución CREG 025 de 1995 (CREG, 1995).

Entre los requisitos de desempeño de telecomunicaciones establecidos por (CREG, 1995), se encuentran la Tabla 8 - 3.

Tabla 8 - 3: Parámetros de desempeño de comunicación según resolución CREG 025 de 1995 (CREG, 1995)

Parámetro de desempeño	Valor
Tasa máxima de error en bits – BER del canal	$1 \cdot 10^{-6}$
Disponibilidad promedio semanal del canal	>97%
Tiempo máximo de desconexión	2 semanas
Resolución del registrador de eventos	1 ms
Disponibilidad anual de las RTU	>98%
Tasa máxima de error – BER en las RTU	$1 \cdot 10^{-5}$

A partir de la información anterior se puede concluir entonces que por cada circuito o bahía de conexión, el bus de proceso debe contemplar:

- Una protección principal
- Una protección de respaldo
- Un registrador de fallas
- Un reloj GPS
- Una *Merging Units* (equivalente a una RTU)
- Cumplir los requisitos de la Tabla 8 - 3

Especificaciones UPME para subestaciones de 500kV

El siguiente análisis se basa en las especificaciones de la subestación Nueva Esperanza 500kV, según (UPME, 2009).

La fabricación de los equipos debe observar, al igual que en las estaciones de 220kV, las normas IEC, ANSI, ITU-T y CISPR. Adicionalmente, se debe cumplir lo siguiente:

- Transformadores de potencia según IEC 60076, con cambiadores de derivación de 21 pasos de 1.25%
- Interruptores según IEC 62271-100
- Los *switches* de red deben cumplir con IEEE 1613 y con IEC 61850-3

En cuanto a los requisitos de telecomunicaciones, también especifica que se deben cumplir las especificaciones de la resolución CREG 025 de 1995, de la Tabla 8 - 3.

En general, las disposiciones de la UPME para una subestación de 500kV son muy similares a los requisitos de una subestación de 230kV, de manera que se puede asumir que para el bus de proceso aplican las mismas condiciones.

Información primaria recopilada en empresas del sector eléctrico

Con el fin de validar los requerimientos antes expuestos, se realizó una entrevista con funcionarios de Empresas Públicas de Medellín. En esta entrevista, se consultó sobre las principales topologías que presentan las subestaciones de esta empresa, y el método que emplean para definir la topología de bus de proceso.

De acuerdo con información suministrada por Profesionales de Proyectos e Ingeniería de EPM, la mayor parte de las subestaciones de la empresa presentan la configuración de doble barra más acople de transferencia o interruptor y medio. Esto coincide con la información encontrada en los diagramas unifilares de la UPME. La Figura 8 - 6 muestra esta configuración en la subestación Sonsón. La figura puede verse más claramente en el Anexo B.

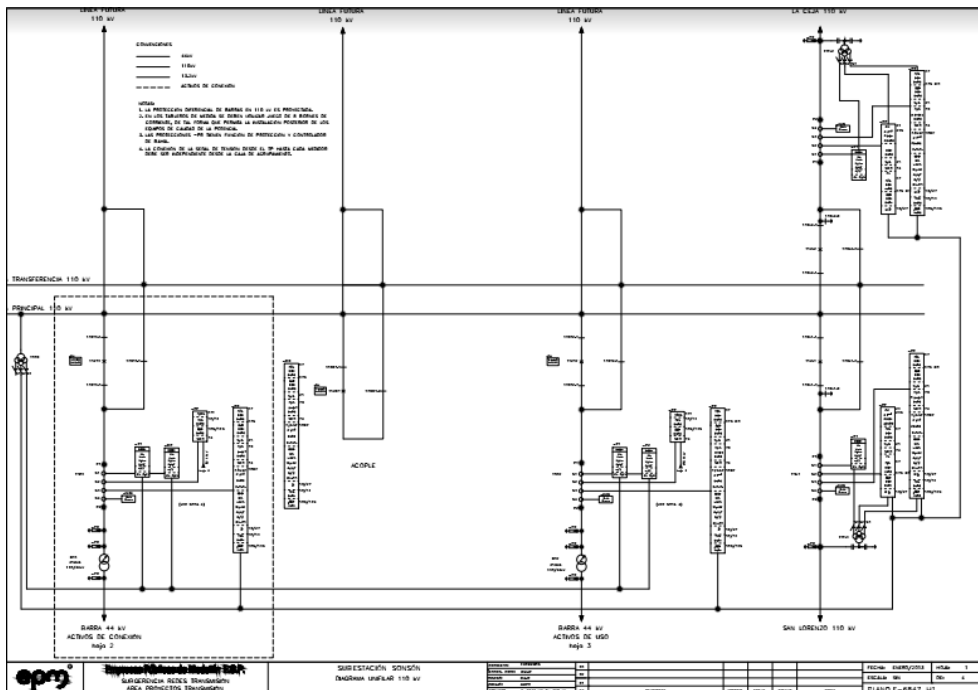


Figura 8 - 6: Diagrama unifilar de la subestación Sonsón (EPM, Anexo B)

En cuanto al método para determinar la topología del bus de proceso, el personal de EPM informó que las arquitecturas tienen en cuenta siempre algún tipo de protocolo redundante, como PRP o HSR. El número de equipos del bus de proceso se define en función del número de bahías de la subestación, debido a que por cada bahía se implementa un controlador y protección secundaria, una protección primaria y dos *Merging Units* análogas. Estos dispositivos se conectan más adelante mediante *switches* al bus de estación. La Figura 8 - 7 muestra un ejemplo de arquitectura PRP de EPM, distinguiendo tres bahías: bahía de transformador, de línea 1 y de línea 2. Se evidencia que los equipos se replican por cada bahía.

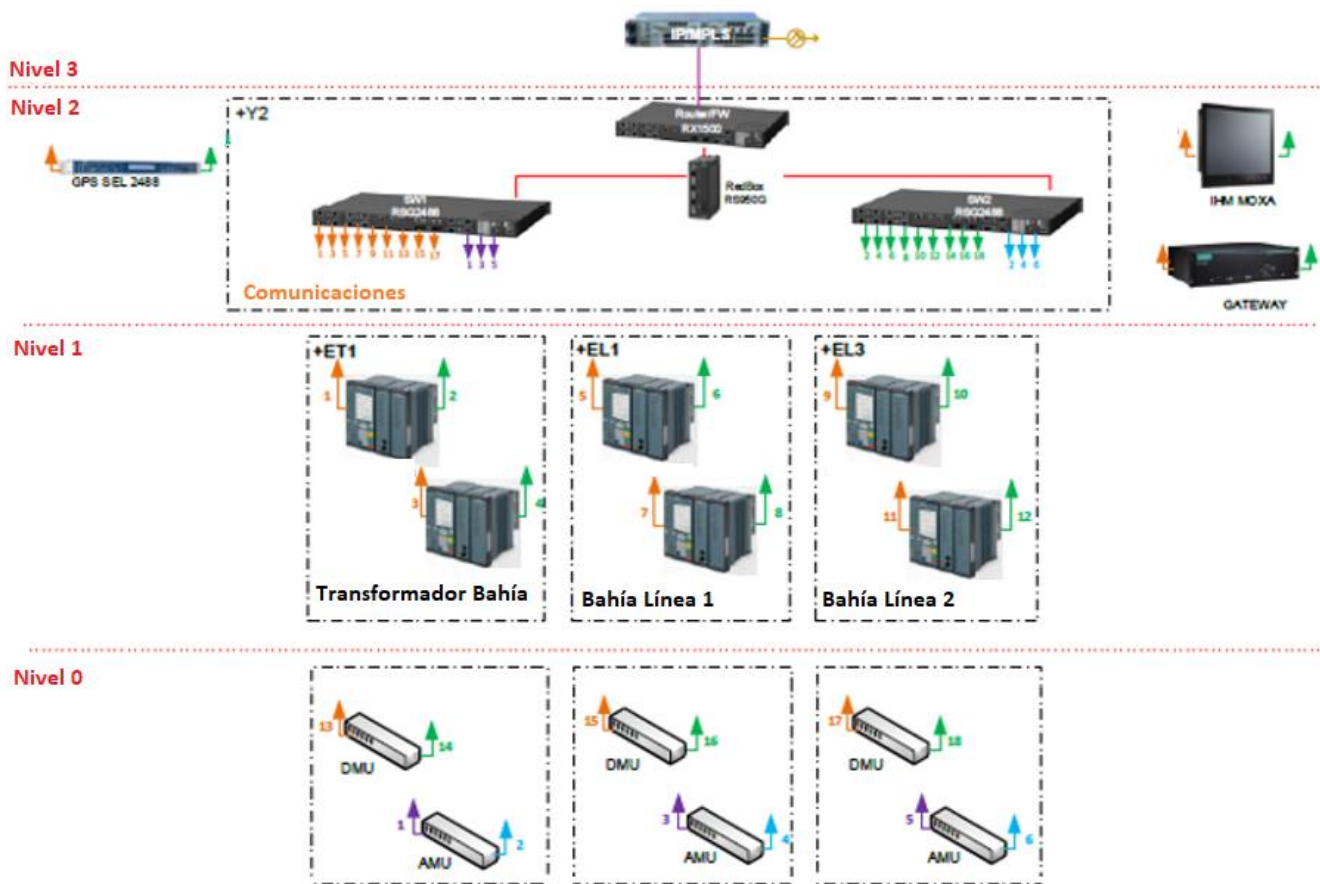


Figura 8 - 7: Ejemplo de topología empleada por EPM (EPM, Anexo B)

Observaciones sobre topologías de bus de proceso

(Barron & Holliday, 2010) presenta la experiencia de la *National Grid Electricity Transmission*, de Reino Unido, en la implementación de buses de proceso en subestaciones. Algunas de las principales recomendaciones de este trabajo son las siguientes:

- La cantidad de equipo primario por bahía (CTs, VTs, interruptores, etc.) dependerá de la configuración de la subestación. Por ejemplo, una subestación en doble barra más acople contendrá más seccionadores por bahía que una subestación de barra sencilla.
- Independientemente de la configuración de la subestación y el número de equipos primarios, la arquitectura del bus de proceso debe permitir el remplazo y mantenimiento de cualquier componente sin interrumpir el servicio. Esto implica redundancia en los componentes o conexiones.
- Independientemente de la configuración de la subestación y el número de equipos primarios, la arquitectura de bus de proceso debe ser estándar, replicable para cada bahía de la subestación.
- Cada bahía debe contener tres IEDs: Protección Principal 1, Protección Principal 2 y Controlador de Bahía. Estos IEDs deben accionar los interruptores de su propia bahía a través del bus de proceso.

Dos topologías que cumplen con los criterios de estandarización y redundancia planteados anteriormente, son propuestas por (Tournier & Werner, 2010): primera corresponde a una doble estrella con redundancia PRP en cada bahía, como se aprecia en la Figura 8 - 8; la segunda corresponde a un anillo con redundancia HSR en cada bahía, como se muestra en la Figura 8 - 9.

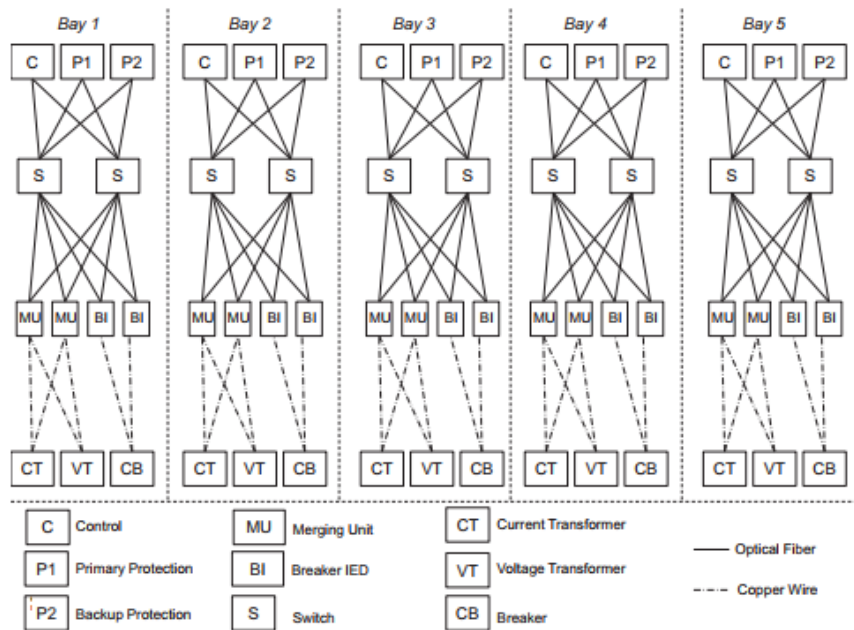


Figura 8 - 8: Arquitectura estándar PRP por bahía (Tournier & Werner, 2010)

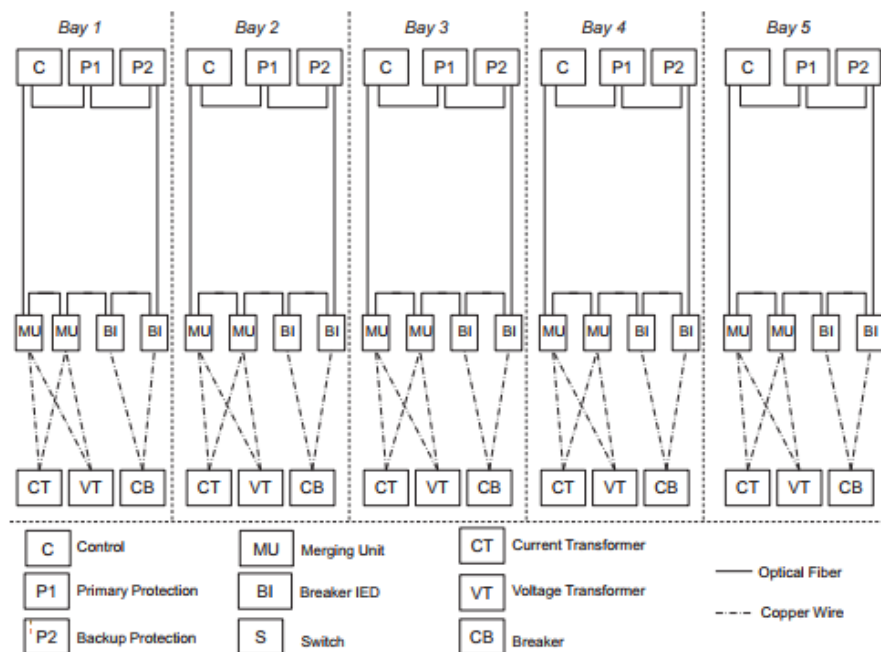


Figura 8 - 9: Arquitectura estándar HSR por bahía (Tournier & Werner, 2010)

Las arquitecturas de la Figura 8 - 8 y la Figura 8 - 9 podrían ser aplicables a las subestaciones del Sistema de Transmisión Nacional, pues si se agrega un reloj GPS cumplirían con los equipos de bus de proceso especificados en (UPME, 2012).

Adicionalmente, cumplen con los criterios de flexibilidad y estandarización por bahía propuestos por (Barron & Holliday, 2010). Por otra parte, para que estas arquitecturas resulten completamente aplicables al SIN, sería necesario que los equipos empleados cumplieran con los límites de desempeño presentados en la Tabla 8 - 3.

En el siguiente apartado de este trabajo, se presenta el desarrollo de un ambiente de simulación para arquitecturas de bus de proceso, el cual hace énfasis en la arquitectura de la Figura 8 - 8. Como se verá, el ambiente de pruebas desarrollado incluye la adición de un reloj GPS en la arquitectura, además de tener en cuenta los parámetros comerciales de los IEDs y las *Merging Units* comerciales, como una aproximación al cumplimiento de lo establecido en la Resolución CREG 025 de 1995 sobre el desempeño de las telecomunicaciones.

9. Desarrollo de la aplicación en OMNET++

9.1. Contexto de la herramienta OMNET++

OMNET++ es un ambiente para la simulación de redes de eventos discretos, orientado a objetos, con una arquitectura genérica que permite su aplicación en el modelado de redes de comunicación, protocolos y colas (Varga & OpenSim Ltd., 2016). OMNET++ no es un simulador de redes de comunicación, sino más bien una plataforma para desarrollar simulaciones de cualquier sistema que pueda modelarse mediante Eventos Discretos. Es, hasta cierto punto, análogo a una aplicación genérica de programación o modelado de ingeniería, aunque sí está muy orientado a sistemas de eventos discretos que puedan interactuar mediante el envío de mensajes. OMNET++ está basado en C++ y es gratuito para uso académico, aunque existe una versión paga para uso comercial, llamada OMNEST, la cual es licenciada por la empresa SimulCraft Inc (Varga, 2003). En el Anexo C se presenta una guía para la construcción de redes en OMNET++, la cual describe las principales funcionalidades que fueron aprovechadas para la realización del presente trabajo.

Debido a que OMNET++ es un entorno de desarrollo general, no cuenta por sí solo con módulos prediseñados para emular protocolos reales de comunicación, como por ejemplo TCP-IP o UDP. Sin embargo, se ha desarrollado una variedad de “marcos externos” para implementar protocolos concretos, facilitando en parte el trabajo de programación mediante el suministro de “plantillas” de paquetes o módulos de red. Algunos de los marcos externos más relevantes son los siguientes:

- INET: es una librería de código abierto, que contiene protocolos y modelos de redes inalámbricas y móviles. Incluye plantillas de protocolos de internet como TCP, UDP, IPv4, IPv6, OSPF y BGP. Incluye también paquetes útiles para modelar protocolos Ethernet y el estándar IEEE 802.11 (INET Team, 2016). INET es concebido como un proyecto colaborativo, se alimenta de los aportes de los usuarios que hacen parte de una comunidad, en la cual cada uno aporta sus códigos y desarrollos. Por esta razón, es un marco en constante desarrollo; su manual de uso se encuentra en borrador.
- INETMANET: es un marco orientado a la simulación de redes móviles Ad Hoc. Una red MANET (*Mobile Ad-hoc Network*) es una red inalámbrica sin infraestructura fija de acceso, en la cual los dispositivos pueden moverse físicamente y conectarse y desconectarse dinámicamente a otros dispositivos que estén cerca, de manera que cada nodo del sistema tiene la capacidad de actuar como enrutador en determinados momentos (Conti & Giordano, 2014). INETMANET es el marco para OMNET++ que contiene modelos para redes Ad-hoc, tales como nodos móviles y control del canal (Noorhayati, Norashidah, Ejaz, & Awang, 2016). La librería puede ser descargada de la página web: <https://github.com/aarizaq/inetmanet-3.x>
- SimuLTE: las redes *Long Term Evolution – LTE* son un estándar UMTS (3GPP-TS 36.300) para redes celulares. SimuLTE es un marco para simulación de redes LTE, desarrollado por el grupo de investigación *Computer Networking Group*, de la Universidad de Pisa.
SimuLTE implementa los datos de las redes de acceso LTE y el núcleo de paquetes empleados por estas redes (Virdis, Stea, & Nardini, 2014). Las librerías de este marco pueden descargarse en la página web <http://simulte.com/index.html>

Pese a la existencia de estos marcos y de otros más, el presente trabajo no empleó ninguno, básicamente por dos razones:

En primer lugar, se requería de cierta flexibilidad adicional, para definir la estructura de los paquetes, las conexiones y la dinámica de la red, de manera que se prefirió programar todos los módulos de la red desde el principio.

En segundo lugar, el ambiente de simulación propuesto por el presente trabajo se concentra más en los parámetros de los dispositivos comerciales (tiempo de procesamiento, número de puertos, etc.) que en la estructura propia de cada protocolo, de manera que no se concedió tanta importancia a los aspectos contenidos en los marcos, como las estructuras TCP-IP.

9.2. Diseño e implementación del ambiente de simulación

A continuación se describe el procedimiento para la construcción del ambiente de simulación de arquitecturas 61850 en OMNET++. Los procedimientos detallados para construir la mayor parte de las funcionalidades del ambiente de simulación se explican en el Anexo C.

9.2.1. Definición de la arquitectura seleccionada

Para cumplir con las recomendaciones de (UPME, 2012), (CREG, 1995) y (Barron & Holliday, 2010), se seleccionó una arquitectura estándar para cada bahía de las subestaciones: la arquitectura de doble estrella con redundancia PRP.

Esta arquitectura tiene la siguiente composición por bahía de conexión:

- Un controlador de bahía
- Una protección principal 1
- Una protección principal 2
- Un registrador de fallas
- Un reloj GPS
- Dos *switches* de red
- Al menos una *Merging Unit*

Adicionalmente, para la configuración de la arquitectura se tuvieron en cuenta las recomendaciones de los investigadores del Laboratorio IEC 61850, de la Universidad Nacional de Colombia, Sede Medellín (<http://grupotytl.medellin.unal.edu.co/laboratorio-iec61850>). Uno de los aportes más significativos que realizó el personal del laboratorio, fue la sugerencia de segmentar el tráfico de la arquitectura PRP en dos redes virtuales o VLANs, las cuales pueden ser activadas o desactivadas desde los *switches* de red.

Esta configuración se muestra en la Figura 9 - 1, resaltando cada VLAN con un color diferente: azul, naranja, rojo y verde. Adicionalmente, el reloj GPS se conecta mediante puerto IRIG-B a las *Merging Units* (color negro) y mediante protocolo PTP a los *switches* de red (color violeta).

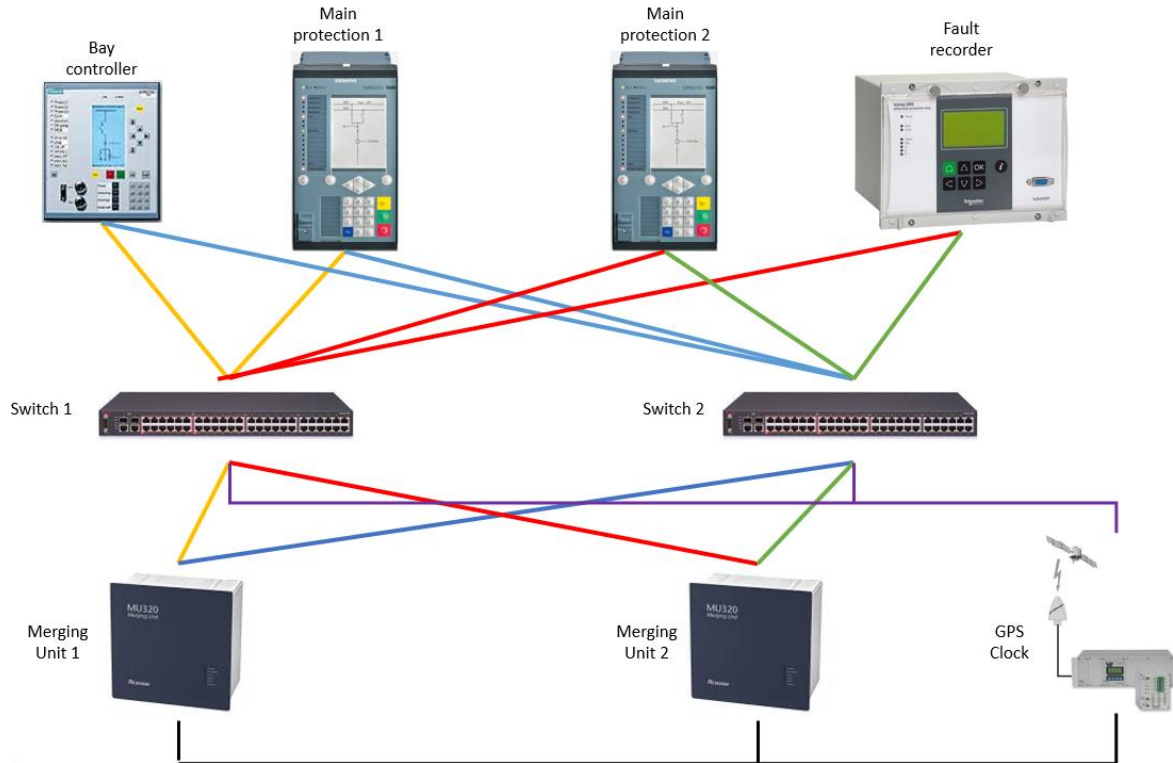


Figura 9 - 1: Arquitectura sugerida por el Laboratorio IEC 61850 – UNAL (elaboración propia, según orientación del Laboratorio IEC 61850 UNAL)

Otro ajuste significativo en la arquitectura PRP de doble estrella por bahía parte de una de las consideraciones de (Barron & Holliday, 2010), según la cual el número de equipos por bahía (CTs, VTs, seccionadores, etc.) puede influir en el número de equipo primario del bus de proceso. Para atender a esto, la arquitectura implementada en OMNET++ se programó de tal manera que pueda implementar un número arbitrario de *Merging Units* por bahía, según criterio del usuario en el momento de la inicialización.

Con las anteriores consideraciones en mente, se procedió a implementar el archivo .ned en OMNET++, para definir la arquitectura del bus de proceso. La Figura 9 - 2 muestra el listado de equipos empleados en la arquitectura, que fueron implementados en OMNET++.



Figura 9 - 2: Listado de equipos usados en la arquitectura (elaboración propia)

Las *Merging Units* de la Figura 9 - 2 fueron implementadas teniendo en cuenta los parámetros de fabricantes recopilados en la Tabla 7 - 5, para las referencias Reason MU320, SAM600CT, SAM600VT, SIIPROTEC 6MU805 y AMU VIZIMAX. Las *Merging Units* ubicadas en la rama 2 de la arquitectura (ver Figura 9 - 1) se distinguen mediante el sufijo "R2". Adicionalmente, se definieron dos equipos genéricos con objeto de pruebas de código, llamados MergingUnit y MergingUnitR2, que no corresponden a ninguna referencia comercial.

En cuanto a los demás equipos (controlador de bahía, registrador de fallas, protecciones principales 1 y 2, *switches* y reloj GPS) se modelaron tomando una referencia comercial aleatoria del listado que va de la Tabla 7 - 1 a la Tabla 7 - 4.

La arquitectura genérica implementada en OMNET++ se muestra en la Figura 9 - 3. La Rama 1 está compuesta por los siguientes equipos:

- Protección principal 1
- Controlador de bahía
- *Switch* 1
- *Merging Units* del lado izquierdo

La Rama 2 está compuesta por los siguientes equipos:

- Protección principal 2
- Registrador de fallas
- *Switch 2*
- *Merging Units* del lado derecho

Las VLANs son representadas mediante colores en las conexiones: color azul para la LAN A de la Rama 1, verde para la LAN B de la Rama 1, rojo para la LAN A de la Rama 2 y violeta para la LAN B de la Rama 2.

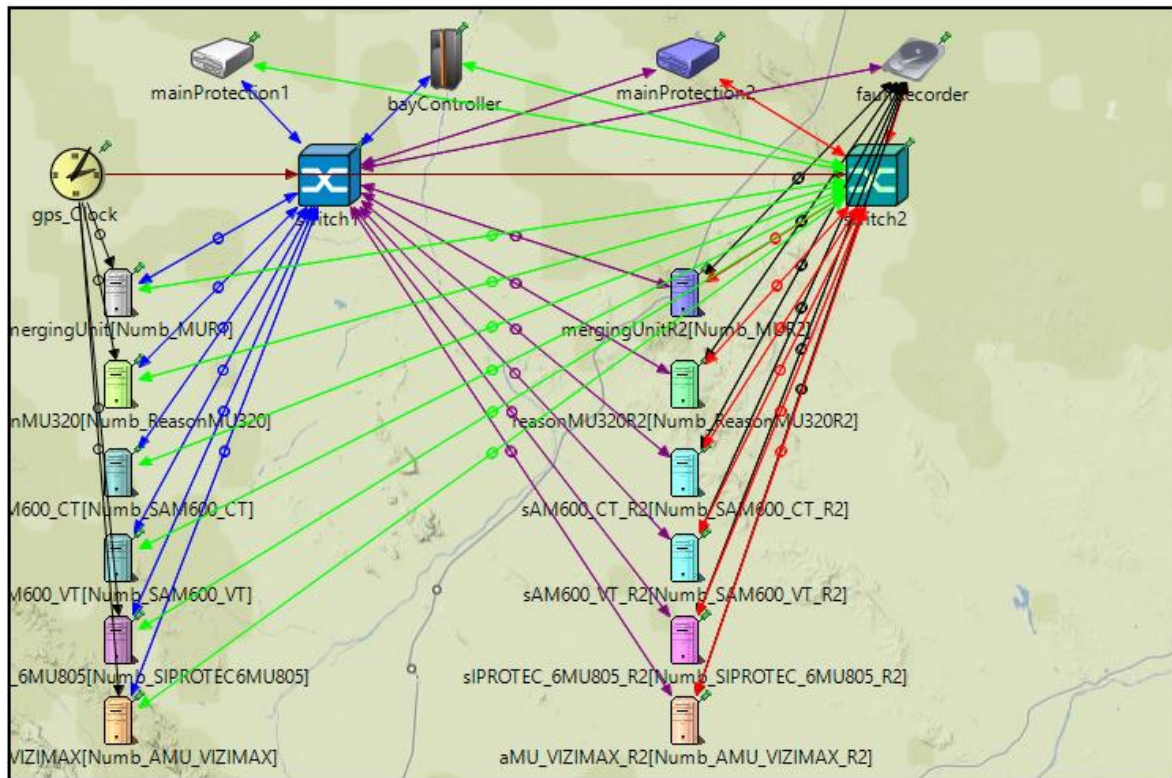


Figura 9 - 3: Arquitectura genérica implementada en OMNET++ (elaboración propia)

Las conexiones de las *Merging Units* a los *switches* aparecen con un círculo en el medio. Esto indica que la conexión no es fija, sino dinámica, dependiente del número de *Merging Units* seleccionadas por el usuario durante la configuración inicial de la red. Así por ejemplo, la Figura 9 - 4 muestra una realización de la arquitectura genérica en la que el usuario ha inicializado la red con una *Merging Units* Reason MU320 en la Rama 1 y una *Merging Units* SAM 600CT en la Rama 2.

Por su parte, la Figura 9 - 5 muestra una configuración inicial de la arquitectura con una ReasonMU 320, dos SAM 600CT y una AMU VIZIMAX en la Rama 1, además de 3 Reason MU320 y dos SAM 600VT en la Rama 2. Esto se logra incluyendo el siguiente código en la definición del archivo .ned:

parameters:

```

    //Merging Units en la Rama 1
    int Numb_MUR1 @prompt("Ingrese el número de Merging Units en la
Rama 1"); //número de Merging Units en la Rama 1
    int Numb_ReasonMU320 @prompt("Ingrese el número de ReasonMU320 en
la Rama 1"); //número de Merging Units en la Rama 1
    int Numb_SAM600_CT @prompt("Ingrese el número de SAM600_CT en la
Rama 1"); //número de Merging Units en la Rama 1
    int Numb_SAM600_VT @prompt("Ingrese el número de SAM600_VT en la
Rama 1"); //número de Merging Units en la Rama 1
    int Numb_SIPROTEC6MU805 @prompt("Ingrese el número de SIPROTEC
6MU805 en la Rama 1"); //número de Merging Units en la Rama 1
    int Numb_AMU_VIZIMAX @prompt("Ingrese el número de AMU VIZIMAX en
la Rama 1"); //número de Merging Units en la Rama 1
    //Merging Units en la Rama 2
    int Numb_MUR2 @prompt("Ingrese el número de Merging Units en la
Rama 2"); //número de Merging Units en la Rama 2
    int Numb_ReasonMU320R2 @prompt("Ingrese el número de ReasonMU320
en la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_SAM600_CT_R2 @prompt("Ingrese el número de SAM600_CT en
la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_SAM600_VT_R2 @prompt("Ingrese el número de SAM600_VT en
la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_SIPROTEC_6MU805_R2 @prompt("Ingrese el número de SIPROTEC
6MU805 en la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_AMU_VIZIMAX_R2 @prompt("Ingrese el número de AMU VIZIMAX
en la Rama 2"); //número de Merging Units en la Rama 2

```

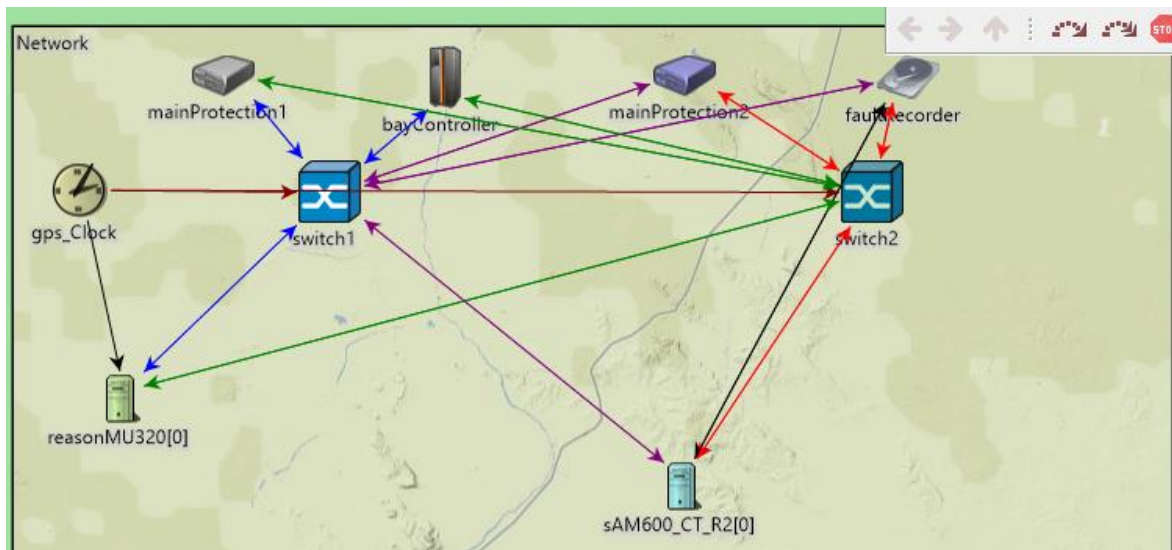



Figura 9 - 4: Arquitectura con una MU en cada rama (elaboración propia)

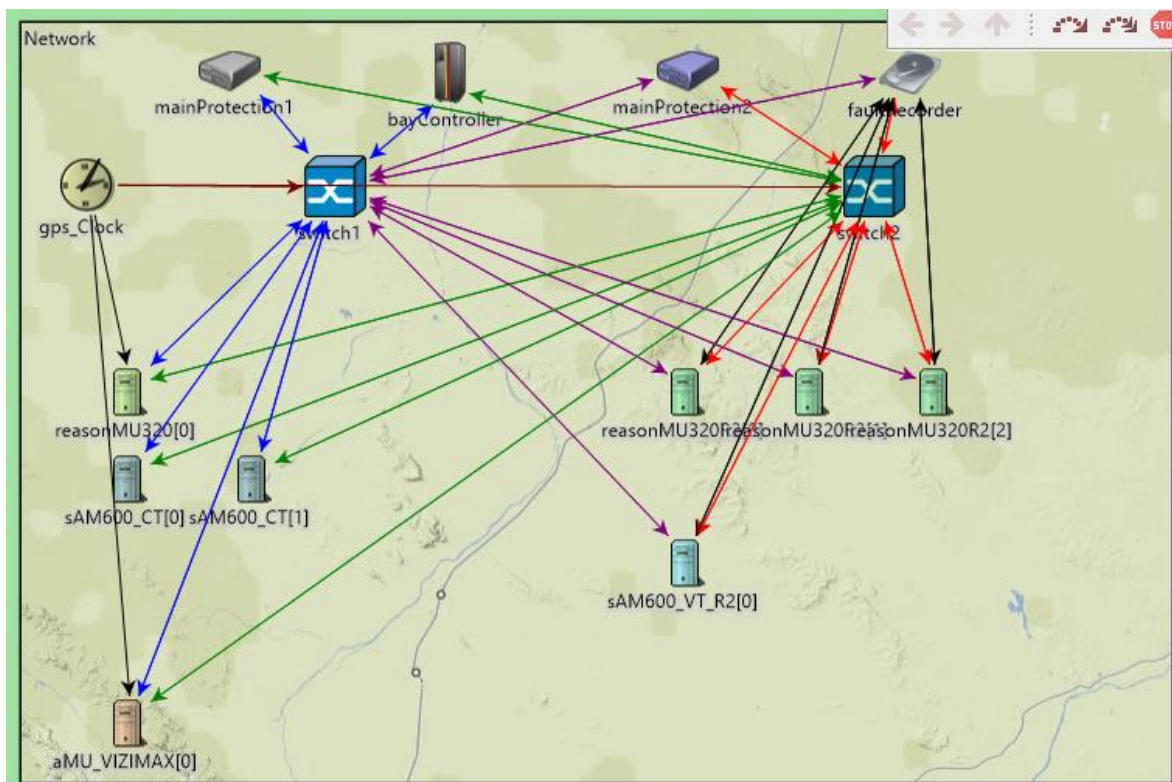


Figura 9 - 5: Arquitectura con diversas MU en cada rama (elaboración propia)

Una vez definido el archivo .ned, se procedió a definir el código de comportamiento de cada módulo de la red, mediante la definición del archivo .cc de cada módulo. Este procedimiento se explica en el siguiente apartado.

9.2.2. Definición de cada módulo de la arquitectura

Para la definición del comportamiento de cada módulo fue necesario considerar la dinámica del tráfico en el bus de proceso, según las indicaciones de la norma IEC 61850, como se explica a continuación.

Dinámica general del tráfico de información en el bus de proceso

En el bus de proceso se presenta el siguiente tráfico:

- Mensajes *Sampled Value* – SV: los *Sampled Values* son definidos en la norma IEC 61850-9-2 (IEC, 2009, pp. 61850–9). Son mensajes enviados por las *Merging Units* a partir de los datos recopilados de los dispositivos de medición. Los valores análogos de tensión y corriente son digitalizados por las *Merging Units* y empaquetados en un mensaje *Sampled Value*. Esto se realiza a una tasa de 80 muestras por ciclo en modo muestreo.
- Mensajes GOOSE: *Generic Object Oriented Substation Event* – GOOSE es una clase de mensaje definido en la norma IEC 61850-7-2 (IEC, 2010, pp. 61850–7–2) e IEC 61850-8-1 (IEC, 2011, pp. 61850–8–1). Estos mensajes son empleados para comunicación punto a punto entre dispositivos, para el reporte de estados y para el envío de órdenes o comandos entre IEDs.
- Mensajes de sincronización: la sincronización del sistema puede ser obtenida empleando un reloj GPS, que envía mensajes de sincronización vía IRIG-B, un protocolo que permite especificar fecha y hora entre dispositivos de subestaciones (De Dominicis, Ferrari, Flammini, Rinaldi, & Quarantelli, 2010).

Otra manera de sincronizar los equipos en la subestación es mediante mensajes *Precision Time Protocol* – PTP, definidos por el estándar IEEE 1588 (IEEE, 2008). Estos mensajes se diseminan por toda la red, con direcciones destino *broadcast* y permiten que todos los dispositivos puedan tomar la hora del mejor reloj disponible en la red.

La dinámica de este tráfico en el bus de proceso es descrita por (Konka et al., 2011): una *Merging Units* toma varias señales de tensión y corriente provenientes de CTs y VTs, y las digitaliza, empaquetándolas en mensajes *Sampled Value*; la *Merging Units* envía el *Sampled Value* a los IEDs de control, a través del switch de red; los IEDs que se encuentran suscritos al servicio *Sampled Value* toman estos mensajes y los analizan en búsqueda de alguna anomalía en las señales muestreadas; en caso de alguna situación de falla, el IED de control envía un comando al interruptor de su bahía, en forma de un mensaje GOOSE; el interruptor recibe este mensaje GOOSE y despeja la falla, además de enviar un nuevo mensaje GOOSE reportando su estado.

El flujo del tráfico descrito anteriormente se puede resumir en la Figura 9 - 6, suministrada por el personal del Laboratorio IEC 61850 de la Universidad Nacional Sede Medellín.

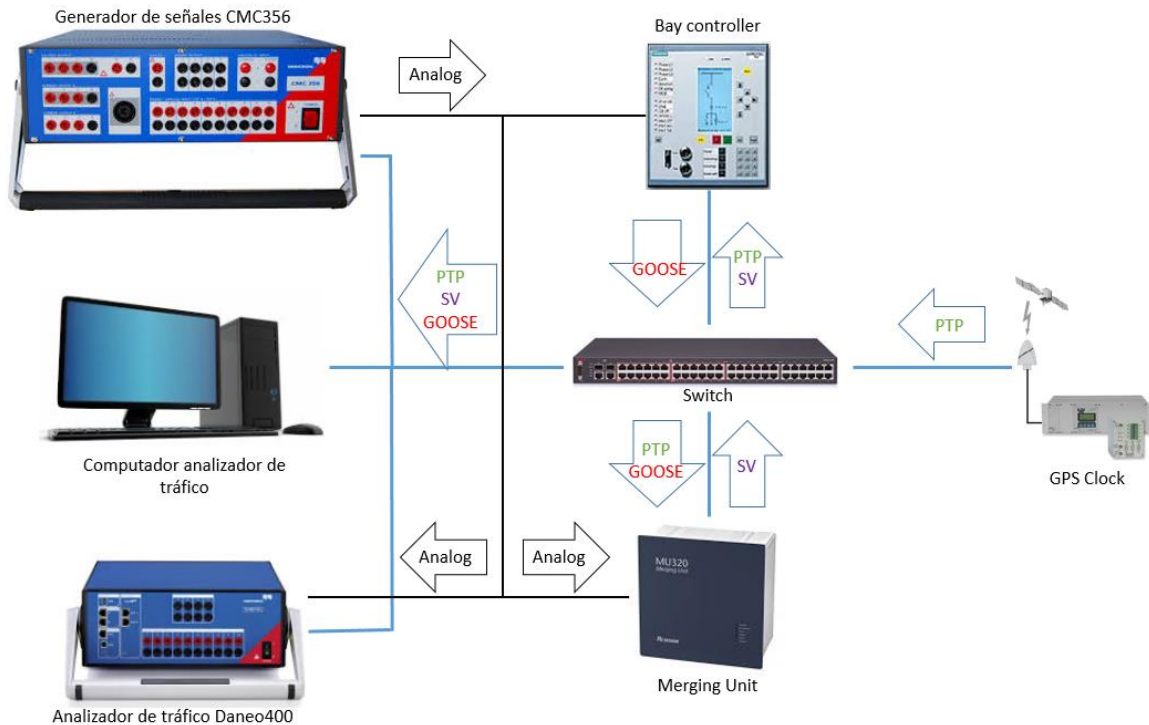


Figura 9 - 6: Dinámica del tráfico de red en el bus de proceso (elaboración propia según orientación del Laboratorio IEC 61850 UNAL)

Con las anteriores consideraciones en mente, se programaron los archivos .cc de cada uno de los módulos de la arquitectura, diferenciando la respuesta de cada módulo ante mensajes *Sampled Value*, GOOSE, IIRIG-B o PTP.

Programación de las *Merging Units*

En primer lugar, se modeló cada *Merging Units* de acuerdo con los parámetros de la Tabla 7 - 5, teniendo en cuenta el número de puertos, el tiempo de procesamiento, la presencia de conexión IRIG-B y la estructura general de un paquete *Sampled Value*.

El siguiente extracto de código corresponde a la declaración de la *Merging Units* de la referencia SAM600 CT:

```
class SAM600_CT : public cSimpleModule
{
protected:
    virtual void initialize() override;
    virtual void handleMessage(cMessage *msg) override;
    int getArrivalGateId() const;
    virtual void finish() override;

    (...)
}
```

Todas las *Merging Units* fueron programadas con los cuatro métodos del código anterior, diferenciándose en los valores de los parámetros de cada método. Estos métodos se explican a continuación.

- Initialize(): este método permite inicializar los valores de las variables y principales parámetros de cada Mergin Unit. Por ejemplo, para la SAM600 CT, que tiene un tiempo de digitalización de 2ms, 4 entradas análogas (corriente trifásica con su neutro) y una velocidad de transferencia de datos de 100Mbps se define el siguiente código de inicialización:

```
void SAM600_CT::initialize()
{
////////// Definición de parámetros SAM600CT //////////
#define DELAY 0.002;//Retardo entre la recepción de la muestra y el
envío del SV: 2ms
#define SPEED "100Mbps";
////////////////////////////////////
/
```

```

seq=1;
QualityBIT="TRUE_";
SynchBIT="TRUE_";
Sequence=std::to_string(seq);
X14_IPH1="INTE_";
X13_IPH2="INTE_";
X12_IPH3="INTE_";
X11_NI="INTE_";
SrcAddress=SAM600_CT;
DstAddress=MULTI;

time=0;

SV=SrcAddress+DstAddress+QualityBIT+SynchBIT+Sequence+X14_IPH1+X13_IP
H2+X12_IPH3+X11_NI;
strcpy(Sample,SV.c_str());
cMessage *start = new cMessage(Sample);
scheduleAt(0.0001, start); //Mensaje para iniciar el tráfico
}

```

Las instrucciones `cMessage *start = new cMessage(Sample);` y `scheduleAt(0.0001, start);` permiten crear un mensaje inicial para iniciar el tráfico de *Sampled Values*. En realidad, consiste en un “automensaje”, que la *Merging Units* se envía a sí misma y que no se ve reflejado en la red. Este es el mecanismo que OMNET++ implementa para iniciar tráfico, ya que el resto de acciones son detonadas a partir de la recepción del primer mensaje.

- `HandleMessage(cMessage *msg)`: esta función ejecuta las acciones requeridas cuando se recibe un mensaje tipo `msg`. Esta rutina funciona así:

Luego de la ejecución de la rutina de inicialización, la *Merging Units* se envía a sí misma un mensaje ficticio. Cuando se recibe este mensaje, se activa la rutina *HandleMessage*. Esta rutina evalúa el contenido del mensaje; si es un “automensaje”, toma una muestra de los valores de tensión y corriente y los empaqueta en un mensaje *Sampled Value*, el cual es enviado inmediatamente por el puerto de salida de la Mergin Unit. Finalmente, se programa un nuevo “automensaje” en los próximos 0.2083 ms, para volver a activar la rutina unas 80 veces por ciclo. Esto se puede apreciar en el siguiente extracto de código:

```
void SAM600_CT::handleMessage(cMessage *msg)
{
    sender=msg->isSelfMessage();
    delete(msg);
    time=simTime().dbl();
    seq++;
    if(sender==1){//recibe automensajes para programar SV a 80
muestras/segundo

        SampledValue *nuevo = new SampledValue();
        nuevo->setNombre("Nuevo Sampled Value");
        nuevo->setSourceAddress(5);//5=SAM600CT
        nuevo->setDestinationAddress("MULTI");
        nuevo->setLAN_name("LAN_A");
        nuevo->setQuality_BIT(1);
        nuevo->setSynch_BIT(1);
        nuevo->setSeq(seq);
        nuevo->setV_PH1(230*sin(2*PI*60*time));
        nuevo->setV_PH2(230*sin(2*PI*60*time+PI/3));
        nuevo->setV_PH3(230*sin(2*PI*60*time+2*PI/3));
        nuevo->setN_V(0);
        nuevo->setI_PH1(100*sin(2*PI*60*time));
        nuevo->setI_PH2(100*sin(2*PI*60*time+PI/3));
        nuevo->setI_PH3(100*sin(2*PI*60*time+2*PI/3));
        nuevo->setN_I(0);

        nuevo->setByteLength(8);
```

```

        sendDelayed(nuevo, 2e-3, "OUT_LAN_A_BRANCH_1"); // retardo
2ms GENERA MUCHA ACUMULACIÓN DE SV
        if(seq < 4800) {
            seq = seq + 1;
        } else {
            seq = 1;
        }
        cMessage *self = new cMessage(Sample);
        scheduleAt(simTime() + 0.2083e-3, self); // emite una
muestra cada 0.2083 ms = 4800 muestras/segundo
    }

```

- finish(): esta rutina se ejecuta al final de las simulaciones, sirve para registrar las señales y para realizar la disposición final de los mensajes.

Adicionalmente, para la operación de los *Sampled Values* fue necesario definir un nuevo tipo de mensaje, que permitiera ensamblar la trama y acceder a la información de forma adecuada por los diferentes módulos suscritos al servicio.

Por ejemplo, la trama *Sampled Value* contiene la dirección del remitente, la dirección del destinatario (generalmente broadcast para que quien quiera se pueda suscribir), un bit de calidad, un bit de sincronización, una secuencia (entero entre 0 y 48000) y la información propiamente dicha de las corrientes y tensiones en cada fase. El siguiente código muestra la definición del paquete *Sampled Value* en OMNET++.

```

packet SampledValue
{
    string Nombre;
    //string SourceAddress;
    int SourceAddress;
    string DestinationAddress;
    string LAN_name;
    int Quality_BIT;
    int Synch_BIT;
    double Seq;
    double V_PH1;
    double V_PH2;

```

```

    double V_PH3;
    double N_V;
    double I_PH1;
    double I_PH2;
    double I_PH3;
    double N_I;
    int PRP_Seq;
}

```

Programación del Controlador de Bahía

El controlador de bahía, al igual que las *Merging Units*, cuenta con las rutinas `initialize()` y `HandleMessage(cMessage *msg)`; sin embargo, esta última rutina ejecuta instrucciones muy diferentes a las *Merging Units*:

- Cuando recibe un *Sampled Value*, revisa los bits de calidad y sincronización. En caso de que ambos bits esté en uno, se procesa el paquete; en caso contrario se descarta el paquete.
- Cuando se procesa el paquete se revisa su contenido, mediante los comandos `pkt1->getV_PHx()` y `pkt1->getI_PHx()`, donde `x` es el número de la fase sobre la cual se quiere revisar el valor de tensión o corriente.
- En caso de alguna anomalía en estos valores, se programa un “automensaje” para el envío de un mensaje GOOSE a las protecciones principales 1 y 2.
- Cuando se recibe un “automensaje”, es porque este corresponde a un mensaje GOOSE programado. Este mensaje GOOSE se envía a las protecciones, para ejecutar la acción de control correspondiente.

Adicionalmente, para la implementación de la redundancia PRP, fue necesario definir un buffer para el controlador de bahía. Este buffer almacena las secuencias PRP de los últimos 10 mensajes recibidos, y permite comparar si un mensaje con la misma secuencia PRP ya fue recibido; en caso afirmativo, se descarta el mensaje. La secuencia PRP es un número aleatorio asignado durante la creación del paquete en la Mergin Unit. El siguiente extracto de código permite visualizar la gestión de los duplicados PRP:

```

if(strcmp("SampledValue",msg->getName())==0) {
    SampledValue *pkt1 = check_and_cast<SampledValue*>(msg);
    source_address=pkt1->getSourceAddress();
    switch(source_address){
    case 1://MU1
        /////// desplazamiento del Buffer
        for(int i=Buffer_size-1;i>0;i--){
            Buffer[i]=Buffer[i-1];
        }
        /////// Ingreso de la nueva secuencia PRP al Buffer
        Buffer[0]=pkt1->getPRP_Seq();
        /////// Identificación de duplicados
        duplicado=0;
        for(int i=1;i<=Buffer_size-1;i++){
            if(Buffer[0]==Buffer[i]){
                duplicado=1;
            }
        }
        break;
    case 2://MU2
        /////// desplazamiento del Buffer
        for(int i=Buffer_size-1;i>0;i--){
            Buffer[i]=Buffer[i-1];
        }
        /////// Ingreso de la nueva secuencia PRP al Buffer
        Buffer[0]=pkt1->getPRP_Seq();
        /////// Identificación de duplicados
        duplicado=0;
        for(int i=1;i<=Buffer_size-1;i++){
            if(Buffer[0]==Buffer[i]){
                duplicado=1;
            }
        }
    }
}

```


Programación de las Protecciones Principales 1 y 2

Las protecciones se programaron también con las funciones `initialize()` y `HandleMessage(cMessage *msg)`. Las protecciones pueden suscribirse al servicio de *Sampled Values*, pero para la arquitectura implementada no responden autónomamente a valores anormales de tensión o corriente.

Mediante la función `HandleMessage(cMessage *msg)`, las protecciones pueden recibir mensajes GOOSE, que pueden indicarles que ha ocurrido una falla y que deben actuar para despejarla. En este caso se simula la actuación de la protección mediante la generación de un nuevo mensaje GOOSE, para el reporte de un cambio de estado en el interruptor controlado. Luego de cierto tiempo, se envía un nuevo mensaje GOOSE reportando que el interruptor ha sido devuelto a su posición inicial, simulando así un recierre posterior a una falla despejada exitosamente.

Programación del Registrador de Fallas

El registrador de fallas se suscribe a los mensajes *Sampled Value* y GOOSE. Cuando el registrador recibe un mensaje GOOSE de parte de una protección, indicando un cambio de estado, se asume que es una falla, y a partir de este momento el registrador guarda en una memoria todos los mensajes *Sampled Value* que lleguen, hasta que se reciba un nuevo mensaje GOOSE notificando la normalización de la situación.

Programación del reloj GPS

El reloj GPS no interactúa bidireccionalmente con ningún módulo de la red. Simplemente genera dos tipos de mensajes de sincronización:

- IRIG-B: se conforma una trama con fecha y hora y se manda un mensaje, 100 veces por segundo, directamente a las *Merging Units* que posean puerto IRIGB
- PTP: es una emulación del *Precisión Time Protocol*. Cada segundo, se empaqueta un mensaje con fecha y hora y se envía con dirección destino broadcast, para que todos los dispositivos de la red puedan leer el mensaje y sincronizarse.

El siguiente extracto de código muestra el envío de estos dos tipos de mensajes de sincronización:

```

void GPS_Clock::handleMessage(cMessage *msg)
{
    //Envío del mensaje por PTP
    if((int)floor(simTime().dbl())>ptpsegundos){
        ptpsegundos++;
        /*ptpreply= msg->getName();
        strcpy(ptpcopy,ptpreply.c_str());
        cMessage *replica1= new cMessage(ptpcopy);
        cMessage *replica2= new cMessage(ptpcopy);
        send(replica1, "OUT_CLOCK_SW1");
        send(replica2, "OUT_CLOCK_SW2");*/
        PTP *sincronizacion1 = new PTP("PTP");
        PTP *sincronizacion2 = new PTP("PTP");
        sincronizacion1->setSeq(simTime().dbl());
        sincronizacion2->setSeq(simTime().dbl());
        send(sincronizacion1, "OUT_CLOCK_SW1");
        send(sincronizacion2, "OUT_CLOCK_SW2");
    }

    // Envío del mensaje por IRIG-B
    send(msg, "OUT_IRIG_B", mergingunits); //como es un vector, el último
    parámetro especifica la puerta de salida
    //sleep(100);
    time=simTime().dbl()+PULSE_RATE;
    Sync_time=std::to_string(time);
    strcpy(tiempo,Sync_time.c_str());
    cMessage *sincronizar = new cMessage(tiempo);
    scheduleAt(simTime()+PULSE_RATE, sincronizar);
}

```

Programación de los switches de red

El principio fundamental de la operación de los *switches* es la identificación de las direcciones fuente y destino y del protocolo del mensaje. Los *switches* contienen las tablas de enrutamiento, con todas las direcciones de todos los dispositivos de la red.

Cuando un *switch* recibe un mensaje, se detecta la dirección del remitente y el tipo de mensaje. Si es un *Sampled Value*, el mensaje se replica en todos los puertos de salida del *switch*, que estén conectados a otros dispositivos que estén suscritos al servicio *Sampled Values*. Si se recibe un mensaje GOOSE, se revisa la dirección destino y se replica el mensaje sólo por el puerto del *switch* conectado a dicho dispositivo destino.

Todo esto se logra mediante la implementación iterativa de casos, como se muestra en el siguiente extracto de código:

```
if(VLAN==0){//filtrado VLAN desactivado
    switch (sender){
    case 1://caso inexistente
        break;
    case 2://GPS Clock
        break;
    case 3://FaultRecorder
        if(strcmp("Please subscribe",msg->getName())==0){
            SV_Enable_PORT6=1;
        }
        if(strcmp("Please unsubscribe",msg->getName())==0){
            SV_Enable_PORT6=0;
        }
        break;
    case 4://BayController
        if(strcmp("Please subscribe",msg->getName())==0){
            SV_Enable_PORT3=1;
        }
        if(strcmp("Please unsubscribe",msg->getName())==0){
            SV_Enable_PORT3=0;
        }
        break;
    case 5://MainProtection2
        if(strcmp("Please subscribe",msg->getName())==0){
            SV_Enable_PORT5=1;
        }
        if(strcmp("Please unsubscribe",msg->getName())==0){
```

```

        SV_Enable_PORT5=0;
    }
    break;
case 6://MainProtection1
    if(strcmp("Please subscribe",msg->getName())==0){
        SV_Enable_PORT2=1;
    }
    if(strcmp("Please unsubscribe",msg->getName())==0){
        SV_Enable_PORT2=0;
    }
    break;
case 7://Switch 1
    break;
case 8://Switch 2
    break;
default://Válido para todas las MergingUnit Rama 1 y Rama 2
    if(SV_Enable_PORT1==1){
        //send(msg,"LAN_A_BRANCH_1_OUTPORT1");//en este puerto están
las MU Rama 1
    }
    if(SV_Enable_PORT2==1){
        send(pkt1,"LAN_A_BRANCH_1_OUTPORT2");//hacia MainProtetion1
    }
    if(SV_Enable_PORT3==1){
        send(pkt2,"LAN_A_BRANCH_1_OUTPORT3");//hacia BayController
    }
    //if(SV_Enable_PORT4==1){
    //    send(replica2,"LAN_B_BRANCH_2_OUTPORT4");//en este puerto
están las MU Rama 2
    //}
    if(SV_Enable_PORT5==1){
        send(pkt3,"LAN_B_BRANCH_2_OUTPORT5");//hacia MainProtection2
    }
    if(SV_Enable_PORT6==1){
        send(pkt4,"LAN_B_BRANCH_2_OUTPORT6");//hacia FaultRecorder
    }
    break;
}

```

9.2.3. Definición de los parámetros definidos por usuario

Con el objetivo de dotar el ambiente de simulación de una mayor flexibilidad, se incluyeron algunos parámetros de la red que pueden ser configurados por el usuario. Los parámetros que se dejan a selección del usuario corresponden al número de equipos por bahía y a opciones de filtrado en los *switches* de red.

Esto se realizó así para atender la recomendación de (Barron & Holliday, 2010) con respecto a la variación de CTs y VTs en función de la configuración de la subestación.

Los parámetros configurables por el usuario, son entonces los siguientes:

- **Número de *Merging Units* por bahía:**

El usuario puede seleccionar, antes de ejecutar la simulación, el número de *Merging Units* de cada referencia, que desea incluir en cada rama. Esto puede ser útil para incluir *Merging Units* redundantes, o para modelar varias bahías en un mismo bus de proceso, concentrando las señales de todos los CTs y VTs en dos únicos *switches* de red.

Lo anterior se logra mediante el siguiente código en la definición del archivo .ned:

parameters:

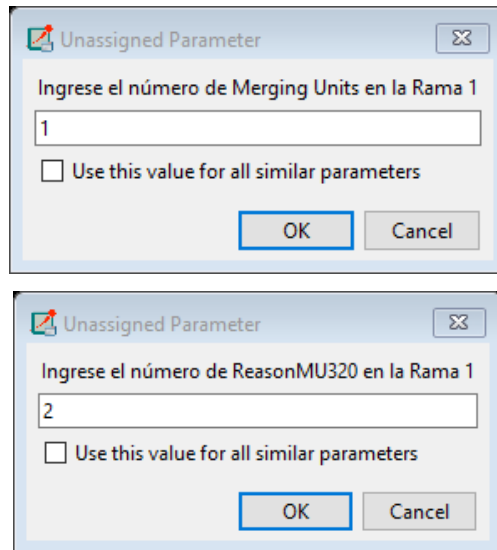
```
//Merging Units en la Rama 1
int Numb_MUR1 @prompt("Ingrese el número de Merging Units en
la Rama 1"); //número de Merging Units en la Rama 1
int Numb_ReasonMU320 @prompt("Ingrese el número de ReasonMU320
en la Rama 1"); //número de Merging Units en la Rama 1
int Numb_SAM600_CT @prompt("Ingrese el número de SAM600_CT en
la Rama 1"); //número de Merging Units en la Rama 1
int Numb_SAM600_VT @prompt("Ingrese el número de SAM600_VT en
la Rama 1"); //número de Merging Units en la Rama 1
int Numb_SIPROTEC6MU805 @prompt("Ingrese el número de SIPROTEC
6MU805 en la Rama 1"); //número de Merging Units en la Rama 1
int Numb_AMU_VIZIMAX @prompt("Ingrese el número de AMU VIZIMAX
en la Rama 1"); //número de Merging Units en la Rama 1
//Merging Units en la Rama 2
```

```

    int Numb_MUR2 @prompt("Ingrese el número de Merging Units en
    la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_ReasonMU320R2 @prompt("Ingrese el número de
    ReasonMU320 en la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_SAM600_CT_R2 @prompt("Ingrese el número de SAM600_CT
    en la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_SAM600_VT_R2 @prompt("Ingrese el número de SAM600_VT
    en la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_SIPROTEC_6MU805_R2 @prompt("Ingrese el número de
    SIPROTEC 6MU805 en la Rama 2"); //número de Merging Units en la Rama 2
    int Numb_AMU_VIZIMAX_R2 @prompt("Ingrese el número de AMU
    VIZIMAX en la Rama 2"); //número de Merging Units en la Rama 2

```

El resultado del código anterior es una serie de cuadros de diálogo mediante los cuales el usuario puede configurar la topología de la red, antes de iniciar la simulación. Por ejemplo, la Figura 9 - 7 muestra tres cuadros de diálogo en los cuales el usuario ha seleccionado un número determinado de *Merging Units* de diferente referencia. La Figura 9 - 8 muestra el resultado arrojado por el ambiente de simulación.



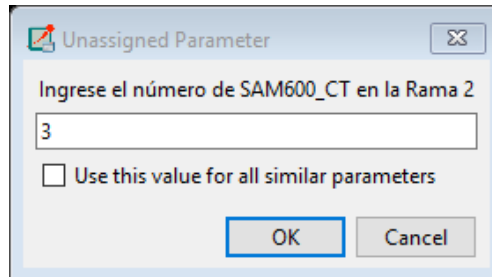


Figura 9 - 7: Cuadros de diálogo de configuración del usuario (elaboración propia)

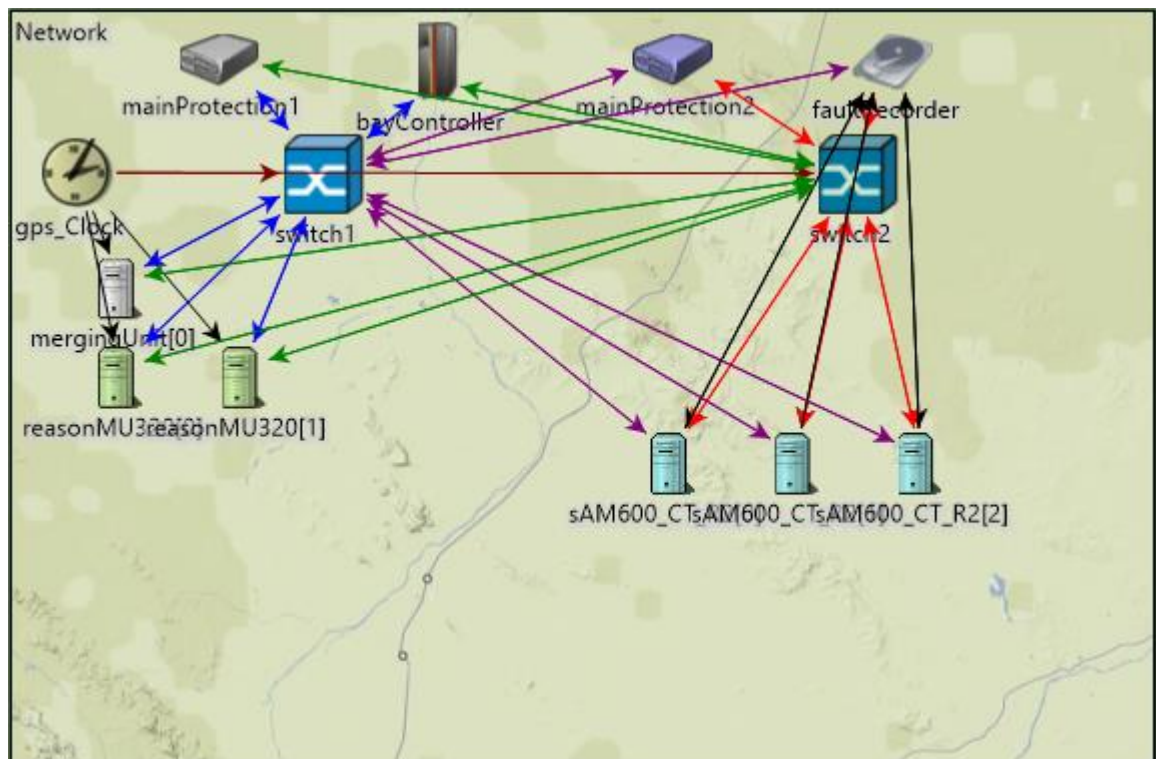


Figura 9 - 8: Topología resultante de la configuración del usuario (elaboración propia)

- **Filtrado de tráfico en los *switches*:**

El usuario puede definir si quiere ejecutar o no una simulación en la cual cada *switch* efectúe un filtrado VLAN de los paquetes. Esto puede ser útil para evaluar los efectos de congestión en la red o para hacer que algunos servicios tengan prioridad o redes dedicadas.

En el ambiente de simulación, las VLAN se distinguen mediante colores en las conexiones: por ejemplo, en la Figura 9 - 8, las conexiones azules representan una VLAN, las conexiones verdes otra VLAN y las violetas otra VLAN. Si se activa el filtrado

VLAN, un mensaje que llegue al *switch* por una conexión azul será replicado únicamente por los demás puertos con conexión azul. Por el contrario, si el filtrado VLAN se desactiva, un mensaje que llegue por una conexión azul será replicado por todos los puertos del *switch*, independientemente de su color.

Esto se logra mediante el siguiente código:

```
int SW1_VLAN @prompt("¿Desea activar el filtrado VLAN en el Switch1?  
1=SI/0=NO");  
int SW2_VLAN @prompt("¿Desea activar el filtrado VLAN en el Switch2?  
1=SI/0=NO");
```

El resultado es una serie de cuadros de diálogo con los cuales el usuario puede activar o desactivar el filtrado VLAN en cada *switch*. La Figura 9 - 9 muestra el cuadro de diálogo para la configuración del filtrado por parte del usuario.

La Figura 9 - 10 muestra el resultado de la desactivación del filtrado VLAN: al lado izquierdo un mensaje llega al *switch* por la conexión azul; al lado derecho, el mensaje es replicado por las conexiones azules, y violetas. La Figura 9 - 11 por su parte muestra el resultado de activar el filtrado VLAN: al lado izquierdo un mensaje llega al *switch* 2 por la conexión verde; al lado derecho, el mensaje es replicado exclusivamente por las conexiones verdes.

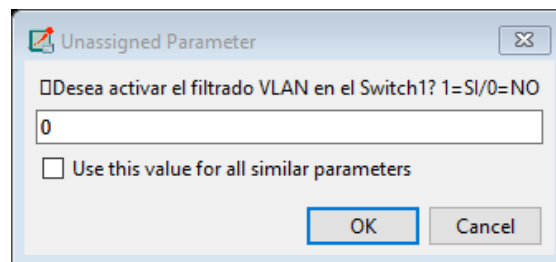


Figura 9 - 9: Configuración de filtrado VLAN por el usuario (elaboración propia)

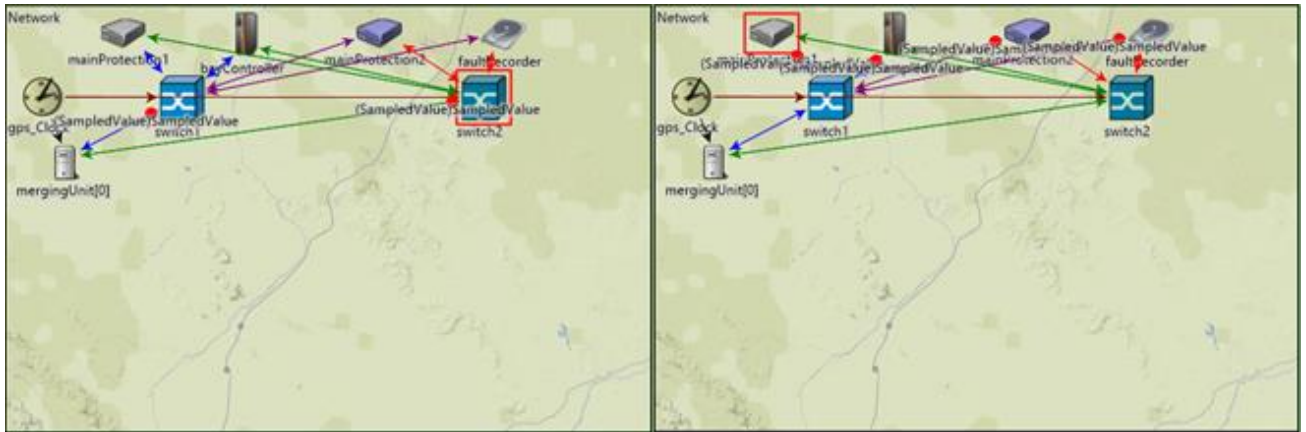


Figura 9 - 10: Filtrado VLAN desactivado (elaboración propia)

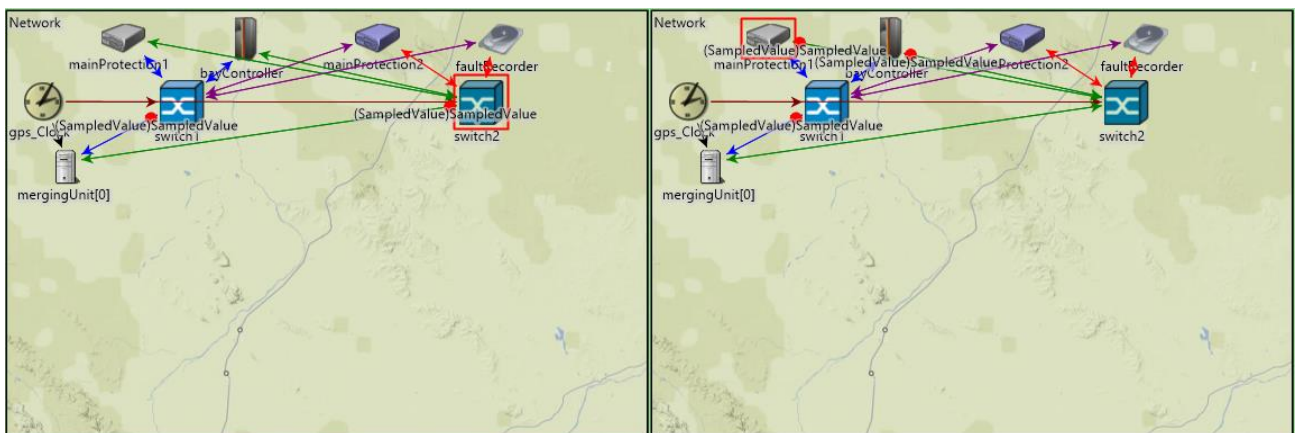


Figura 9 - 11: Filtrado VLAN activado (elaboración propia)

- **Suscripción de los IEDs a diferentes servicios:**

El usuario también puede utilizar el ambiente de simulación para configurar a su voluntad la suscripción de los IEDs a determinados servicios. Por ejemplo, un controlador de bahía puede suscribirse o no a la recepción de *Sampled Values* o de mensajes GOOSE.

Para lograr esto, se requiere coordinar la configuración entre el IED y el *switch* de red: el IED se configura de tal forma que, si el usuario selecciona la opción de suscribir, al iniciar la simulación se envía un mensaje al *switch* solicitando la suscripción al servicio. Luego de esto, el *switch* sólo enviará mensajes por los puertos que se encuentren suscritos al mensaje correspondiente.

La Figura 9 - 12 muestra el instante en que llegan a los dos *switches* los mensajes de solicitud de suscripción a *Sampled Values*, por parte del controlador de bahía y de la protección principal 2.



Figura 9 - 12: Mensajes de solicitud de suscripción a SV (elaboración propia)

9.2.4. Definición de las señales de registro para simulaciones

El principal objetivo del ambiente de simulación desarrollado es evaluar el desempeño de una arquitectura de bus de proceso que sea implementada a partir de un número determinado de dispositivos de diferentes fabricantes.

De acuerdo con (Kumar et al., 2015) y (He et al., 2014), la evaluación del desempeño del bus de proceso se efectúa midiendo los retardos en la transmisión de paquetes y la pérdida de paquetes. Por esta razón, el ambiente de simulación desarrollado contempla la medición de retardos entre el envío y la recepción de paquetes, así como la tasa de pérdida de paquetes. Para realizar dicho registro, fue necesario crear señales en los archivos .ned y .cc, para que alimentaran el archivo .anf con las estadísticas correspondientes. Este proceso se explica a continuación.

Tasa de pérdida de paquetes

Para evaluar la tasa de pérdida de paquetes se requiere realizar un conteo de los paquetes enviados por cada módulo y de los paquetes recibidos por los demás. Debido a la redundancia PRP, se hace necesario definir una señal de conteo de paquetes enviados por cada rama y otra señal para conteo de paquetes recibidos y duplicados.

Las señales de registro de mensajes enviados se declaran en el archivo .ned, en la parte correspondiente a la definición de cada módulo de la red. El código siguiente muestra un ejemplo para la Mergin Unit genérica.

```

simple MergingUnit
{
    parameters:
        @signal[sent] (type="long"); //señal para manipular en el archivo
        .cc Alimenta la señal de conteo real
        @statistic[sentFrames] (source="sent"; record=last; title="sent
        frames"); //Señal alimentada a partir de la que se manipula en el archivo
        .cc Es necesaria para visualizar en el archivo .anf

        @signal[sentR2] (type="long"); //señal para manipular en el
        archivo .cc Alimenta la señal de conteo real
        @statistic[sentFramesR2] (source="sentR2"; record=last;
        title="sent frames R2"); //Señal alimentada a partir de la que se manipula
        en el archivo .cc Es necesaria para visualizar en el archivo .anf

```

Posteriormente, en el archivo .cc, esta señal debe ser activada luego del envío de cada paquete, de manera que se incremente un contador para llevar el registro correspondiente, como se muestra a continuación

```

send(replica_nuevo, "OUT_LAN_B_BRANCH_1");
emit(sentSignalR2, ++sentCounterR2);

```

De manera similar, se pueden programar señales análogas en los IEDs de protección y control, para activarlas durante la recepción de los mensajes, identificando así el número de paquetes recibidos.

El resultado de este código es una tabla que OMNET++ llena automáticamente al finalizar una simulación, contando los paquetes enviados y recibidos por cada uno de los módulos de la red, sobre cada una de las ramas. La Figura 9 - 13 muestra una ventana del archivo .anf, que guarda estos resultados de conteo de paquetes. Los campos que contienen NaN indican que durante la simulación el módulo correspondiente no envió ningún paquete.

Folder	File name	Config na...	R	Run id	Module	Name	Value
/IEC618...	General-0.sca	General	0	General-0-201...	Network.gps_Clock	sentFrames:last	NaN
/IEC618...	General-0.sca	General	0	General-0-201...	Network.faultRecorder	duplicatedFrames:last	9603.0
/IEC618...	General-0.sca	General	0	General-0-201...	Network.faultRecorder	receivedFrames:last	9601.0
/IEC618...	General-0.sca	General	0	General-0-201...	Network.faultRecorder	sentFrames:last	NaN
/IEC618...	General-0.sca	General	0	General-0-201...	Network.bayController	duplicatedFrames:last	9603.0
/IEC618...	General-0.sca	General	0	General-0-201...	Network.bayController	receivedFrames:last	9601.0
/IEC618...	General-0.sca	General	0	General-0-201...	Network.bayController	sentFrames:last	NaN
/IEC618...	General-0.sca	General	0	General-0-201...	Network.mainProtection2	duplicatedFrames:last	9603.0
/IEC618...	General-0.sca	General	0	General-0-201...	Network.mainProtection2	receivedFrames:last	9601.0
/IEC618...	General-0.sca	General	0	General-0-201...	Network.mainProtection2	sentFrames:last	NaN
/IEC618...	General-0.sca	General	0	General-0-201...	Network.mainProtection1	duplicatedFrames:last	9603.0
/IEC618...	General-0.sca	General	0	General-0-201...	Network.mainProtection1	receivedFrames:last	9601.0
/IEC618...	General-0.sca	General	0	General-0-201...	Network.mainProtection1	sentFrames:last	NaN
/IEC618...	General-0.sca	General	0	General-0-201...	Network.switch1	sentFrames:last	NaN

Figura 9 - 13: Conteo de paquetes enviados y recibidos (elaboración propia)

Retardos en el envío de paquetes

El tiempo que transcurre entre el envío de un paquete y la recepción del mismo es una medida del desempeño de la red: cuanto menos sea dicho tiempo más eficiente será la red. En el caso del ambiente de simulación para arquitecturas de bus de proceso, este tiempo se ve influenciado por los tiempos de procesamiento de cada módulo, por los retardos de cada canal y por el número de conexiones por las que pasa el paquete.

Los tiempos de procesamiento de las *Merging Units*, por ejemplo, fueron tomados de la Tabla 7 - 5. Los retardos en los canales fueron asumidos en 0.1ms y el número de conexiones para cualquier *Sampled Value* es siempre 2, debido a que es necesario pasar por los *switches* de red. De esta manera, el principal determinante del retardo es el tiempo de procesamiento de cada módulo.

El siguiente código muestra el modelamiento de una canal con capacidad de 1Mbps

```
mergingUnit[i].OUT_LAN_A_BRANCH_1 --> { @display("ls=blue");
datarate=1Mbps;} --> switch1.LAN_A_BRANCH_1_INPORT1[i]; //delay = 0.1ms;
```

El siguiente código muestra el modelamiento de un canal con un retardo de 0.1ms

```

sAM600_CT[i-Numb_MUR1-Numb_ReasonMU320].OUT_LAN_A_BRANCH_1    -->    {
@display("ls=blue");          delay          =          0.1ms;          }    -->
switch1.LAN_A_BRANCH_1_INPORT1[i];

```

Para lograr el registro del retardo, simplemente es necesario capturar la estampa de tiempo del paquete cuando fue creado y cuando fue recibido por el módulo correspondiente. Por ejemplo, el paquete *Sampled Value* contiene un campo llamado TIME_stamp, el cual se llena con el tiempo de simulación cuando el mensaje es creado:

```
nuevo->setTIME_stamp(time);
```

Por otro lado, en el archivo .cc del IED de control o protección, se debe definir un vector, para almacenar los tiempos de retardo.

```
cOutVector endToEndDelayVec;
```

Este vector se llena dinámicamente, cada que se recibe un mensaje, con la resta entre el tiempo de simulación actual y la estampa de tiempo en la cual fue creado el mensaje:

```

simtime_t eed = simTime() - pkt1->getTime_stamp();
endToEndDelayVec.record(eed);

```

El resultado es un vector cuya dimensión es el número de mensajes recibidos y que contiene los retardos de cada uno de dichos mensajes. OMNET++ permite visualizar el valor medio, la desviación estándar y la varianza de todos los componentes de dicho vector, como se muestra en la Figura 9 - 14.

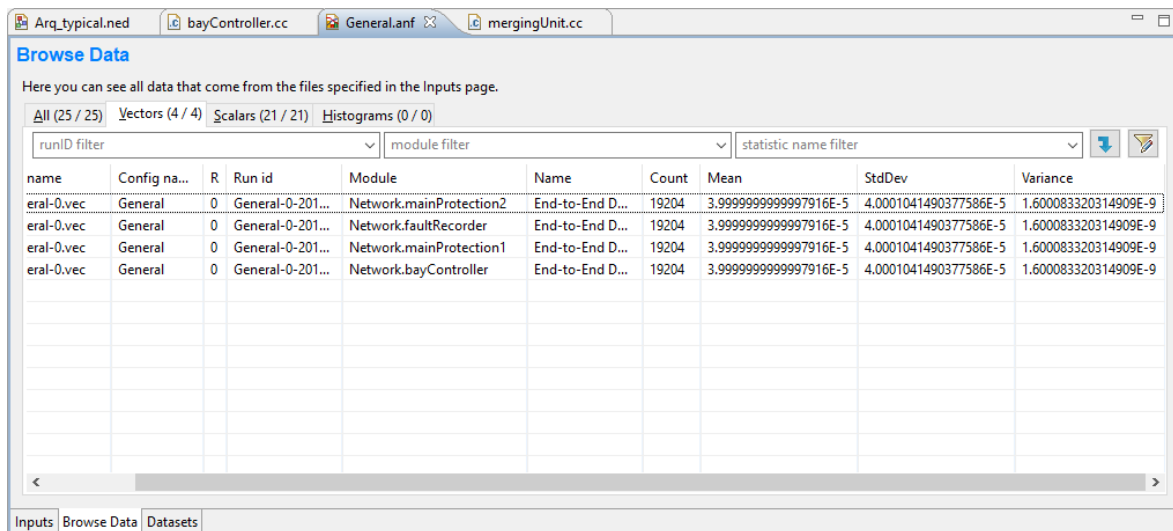


Figura 9 - 14: Resultados de los retardos en algunos IEDs (elaboración propia)

9.3. Simulaciones y resultados

Con el objetivo de poner a prueba la funcionalidad del ambiente de simulación para arquitecturas de bus de proceso, se realizaron una serie de simulaciones de varios casos, para evaluar el desempeño de la red ante diferentes configuraciones de la arquitectura. Estos casos y sus resultados se listan a continuación.

9.1.1 Caso 1: *Merging Units* de una sola referencia sin filtrado VLAN

En este caso, se ejecutaron varias rondas de simulación de 1 segundo de duración (es decir, se deja corriendo el software hasta que el completen los paquetes que se enviarían realmente en el transcurso de un segundo). En cada ronda, se configuró la red para que contuviera sólo una *Merging Units* de una determinada referencia en cada rama. Adicionalmente, se desactivó el filtrado VLAN en los dos *switches*. Los resultados se muestran en la Tabla 9 - 1. Las columnas Envíos R1 y R2 muestran los paquetes enviados por las *Merging Units* en cada rama, y las columnas Recepciones y Duplicados muestran los paquetes recibidos (o reconocidos como duplicados según PRP) en el controlador de bahía. La tasa de pérdida fue calculada como la razón entre las recepciones y la suma de los envíos R1 y los envíos R2.

Tabla 9 - 1: Resultados de simulación para el caso 1 (elaboración propia)

VLAN SW1	VLAN SW2	MU	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida
No	No	Genérica	4801	4801	9601	9603	3.99E-05	0.0104%
No	No	ReasonMU320	4801	4801	9562	9561	4.22E-03	0.4166%
No	No	SAM600CT	4801	4801	9582	9582	2.02E-03	0.2083%
No	No	SAM600VT	4801	4801	9582	9582	2.02E-03	0.2083%
No	No	SIPROTEC6MU805	4801	4801	9548	9547	5.62E-03	0.5624%
No	No	AMU VIZIMAX	4801	4801	9600	9600	1.25E-04	0.0208%

En la Tabla 9 - 1 se puede apreciar que la *Merging Units* que produce el mayor retardo es la SIPROTEC 6MU805. Sin embargo, este valor está muy por encima de los 3ms máximos recomendados por la IEC 61850, de manera que se sospecha que el parámetro de retardo obtenido de catálogos del fabricante podría estar errado. La *Merging Units* comercial con menor retardo sería la AMU VIZIMAX, ya que la *Merging Units* genérica no existe comercialmente sino que fue modelada con propósitos académicos.

Se observa también que el número de paquetes enviados en cada rama es 4801, lo cual se acerca mucho a la tasa de 80 muestras/ciclo recomendada por la norma, ya que $80\text{muestras/ciclo} \times 60\text{Hz} \times 1\text{s} = 4800\text{muestras}$.

La tasa de pérdida fue calculada como la división entre la columna Recepciones y la suma de las columnas Envío R1 y Envío R2. La tasa de pérdida de las *Merging Units* AMU VIZIMAX, SAM 600CT y SAM 600VT es muy similar.

9.1.2 Caso 2: Filtrado VLAN activado

Este caso es exactamente igual al anterior, a excepción de la activación del filtrado VLAN en los dos *switches*. Los resultados se muestran en la Tabla 9 - 2.

Tabla 9 - 2: Resultados del caso 2 (elaboración propia)

VLAN SW1	VLAN SW2	MU	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida
Si	Si	Genérica	4801	4801	4801	4081	4.00E-05	50.0000%
Si	Si	ReasonMU320	4801	4801	4780	4781	4.25E-03	50.2187%
Si	Si	SAM600CT	4801	4801	4790	4792	2.05E-03	50.1146%
Si	Si	SAM600VT	4801	4801	4790	4792	2.05E-03	50.1146%
Si	Si	SIPROTEC6MU805	4801	4801	4774	4773	5.65E-03	50.2812%
Si	Si	AMU VIZIMAX	4801	4801	4800	4800	1.50E-04	50.0104%

El cambio más notable es la reducción a la mitad del número de paquetes recibidos y etiquetados como duplicados. Esto se debe a que cada *Merging Units* está conectada a dos VLANs, de manera que cuando el filtrado VLAN está desactivado, el *Sampled Value* llega por ambas VLANs al destino, mientras que cuando está activado el filtrado, el paquete sólo llega por una VLAN a su destino.

Se observa que la tasa de pérdida se incrementó a niveles cercanos al 50%, pero esto se debe a que se calculó como $\text{Recepciones}/(\text{Envíos R1}+\text{Envíos R2})$. Si se hubiera calculado como $(\text{Recepciones}+\text{Duplicados})/(\text{Envíos R1}+\text{Envíos R2})$ se habrían obtenido valores cercanos a 0% de pérdidas. Ambas ecuaciones son análogas a la ecuación de pérdida de paquetes explicada en el apartado 5.2 Evaluación del desempeño de arquitecturas de comunicación, la cual se basa en el trabajo de (Aranda, 2014); la diferencia está en considerar el término # de paquetes recibidos como igual al número de Recepciones solamente, o como el número de Recepciones + Duplicados.

En cuanto al retardo, se observa que en comparación con la Tabla 9 - 1 cada referencia de *Merging Unit* ha incrementado levemente el retardo. Sin embargo, este incremento en el retardo es tan marginal (es del orden del 1%) que no puede concluirse que se deba a la activación del filtrado VLAN.

9.1.3 Caso 3: Combinaciones entre *Merging Units* de distinta referencia

Otro caso evaluado fue la ocurrencia de una *Merging Units* de distinta referencia en cada rama del bus de proceso. Se ejecutaron rondas de simulaciones de un segundo de duración, con el filtrado VLAN activado en ambos *switches*, combinando la Reason MU320 en la rama 1 con otras referencias de *Merging Units* en la rama 2. Los resultados se muestran en la Tabla 9 - 3.

Tabla 9 - 3: Resultados del caso 3 (elaboración propia)

<i>Merging Units</i>	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa duplicados	Tasa Pérdida
ReasonMU320 - SAM600CT	4801	4801	4780	4780	4.35E-03	50.2187%	0.437%
ReasonMU320 - SAM600VT	4801	4801	4780	4780	4.35E-03	50.2187%	0.437%
ReasonMU320 - SIPROTEC 6MU805	4801	4801	4780	4780	5.02E-03	50.2187%	0.437%
ReasonMU320 - AMU VIZIMAX	4801	4801	4780	4780	4.34E-03	50.2187%	0.437%

Se observa que el retardo se modifica con respecto al de la Tabla 9 - 3, debido a la acción conjunta de las *Merging Units* cada una con distintos tiempos de procesamiento. El ambiente de simulación desarrollado podría emplearse para evaluar cuáles combinaciones de dispositivos de múltiples proveedores funcionan mejor: en este caso, la combinación que produjo el menor retardo fue Reason MU320 y AMU VIZIMAX.

9.1.4 Caso 4: variación del número de *Merging Units* en cada rama

Otro aspecto interesante para evaluar es cómo se ve afectado el desempeño del bus de proceso ante el ingreso de nuevas *Merging Units*, lo cual podría ocurrir por ejemplo si aumenta el número de CTs o VTs, debido por ejemplo al ingreso de una nueva bahía a la subestación.

Para evaluar esto, se ejecutaron rondas de simulación de un segundo de duración, con el filtrado VLAN desactivado en los dos *switches*, conectando 2, 5 y 10 *Merging Units* de la misma referencia en cada rama del bus de proceso. Se ejecutó una ronda de simulaciones para cada referencia de *Merging Units*, con el fin de determinar cuál referencia es más sensible ante cambios en el número de dispositivos.

La Tabla 9 - 4 muestra los resultados para la *Merging Units* genérica, la Tabla 9 - 5 para la Reason MU 320, la Tabla 9 - 6 para las *Merging Units* SAM600CT y SAM 600VT (que obtuvieron idénticos resultados), la Tabla 9 - 7 para la SIPROTEC 6MU805 y la Tabla 9 - 8 para la AMU VIZIMAX.

En la columna “Tasa Pérdida” de las anteriores tablas se observa que al incrementar el número de *Merging Units* se incrementa el porcentaje de paquetes perdidos. Éste fue calculado como $(\text{Recepciones} + \text{Duplicados}) / (\text{Envíos R1} + \text{Envíos R2})$. De nuevo, esta ecuación es análoga a la ecuación de pérdida de paquetes presentada en el apartado 5.2 del presente trabajo (Aranda, 2014), tomando el # paquetes recibidos como Recepciones + Duplicados y el # paquetes enviados como Envíos R1 + Envíos R2.

Aunque no es posible apreciarlo en la columna “Retardo”, el incremento de *Merging Units* también aumentó el tiempo de retardo de los paquetes. Esto se pudo apreciar en el software OMNET++, evidenciando incrementos del orden de $1 \cdot 10^{-9}$ segundos.

Tabla 9 - 4: Resultados del caso 4 para la MU Genérica (elaboración propia)

MU/rama	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida
2	9602	9602	19202	19204	1.07997E-03	0.0104%
5	24005	24005	48002	48013	1.07997E-03	0.0167%
10	48010	48010	95992	96038	1.07997E-03	0.0292%

Tabla 9 - 5: Resultados del caso 4 para la Reason MU320 (elaboración propia)

MU/rama	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida
2	9602	9602	19119	19121	4.32100E-03	0.4426%
5	24005	24005	47793	47807	4.32099E-01	0.4520%
10	48010	48010	95576	95624	4.32099E-01	0.4624%

Tabla 9 - 6: Resultados del caso 4 para la SAM600CT y SAM600VT (elaboración propia)

MU/rama	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida
2	9602	9602	19163	19163	2.12100E-03	0.2135%
5	24005	24005	47903	47912	2.12100E-03	0.2229%
10	48010	48010	95800	95830	2.12100E-03	0.2291%

Tabla 9 - 7: Resultados del caso 4 para la SIPROTEC6MU805 (elaboración propia)

MU/rama	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida
2	9602	9602	19093	19093	5.72010E-03	0.5780%
5	24005	24005	47728	47737	5.72010E-03	0.5874%
10	48010	48010	95447	95483	5.72010E-03	0.5968%

Tabla 9 - 8: Resultados del caso 4 para la AMU VIZIMAX (elaboración propia)

MU/rama	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida
2	9602	9602	19199	19201	2.21000E-04	0.0260%
5	24005	24005	47993	48007	2.21000E-04	0.0354%
10	48010	48010	95976	96024	2.21000E-04	0.0458%

Las anteriores tablas son interesantes porque permiten definir cuál referencia de *Merging Units* es más sensible a la variación en el número de dispositivos: si se evalúa la variación porcentual de la tasa de pérdida entre los casos de 2 y 10 *Merging Units* por rama, se puede medir la sensibilidad.

Por ejemplo, para el caso de la *Merging Units* genérica de la Tabla 9 - 4, la tasa de pérdida con 2 *Merging Units* por rama fue 0.4426%, mientras que para 10 *Merging Units* fue 0.4624%, lo cual representa un incremento del 4% en la tasa de pérdida.

Para el caso de la SAM600CT y SAM600VT se obtiene un incremento del 7% en la tasa de pérdida, al pasar de 2 a 10 *Merging Units* por rama; para la SIPROTEC 6MU805 la tasa de pérdida se incrementó en un 3%; y finalmente, para la AMU VIZIMAX la tasa de pérdida se incrementó en un 76%.

Evidentemente, la AMU VIZIMAX es la *Merging Units* con mejor desempeño temporal, pero es también la más sensible a cambios en el número de dispositivos, incrementando considerablemente la tasa de pérdida ante incrementos en el número de CTs y VTs.

9.1.5 Caso 5: simulación de pérdidas de enlace o corrupción de datos

Uno de los aspectos más importantes de la topología del bus de proceso es la redundancia en los enlaces, debido a que ante una falla en algún puerto o enlace de comunicación es necesario que la comunicación no se interrumpa, para que no ocurran situaciones peligrosas para la subestación y el sistema.

La arquitectura implementada en el ambiente de simulación corresponde a una doble estrella con redundancia PRP. Es importante probar entonces cómo se comporta cuando se pierde uno de los enlaces de comunicación. Para ello, se modifica el parámetro Bit Error Rate – BER, o Packet Error Rate – PER en el canal de comunicaciones a interrumpir.

Por ejemplo, la conexión entre las *Merging Units* genéricas y el *switch* 1 se define mediante el siguiente código:

```
mergingUnit[i].OUT_LAN_A_BRANCH_1 --> { @display("ls=blue"); datarate = 1Mbps; per=0.1e-3;} --> switch1.LAN_A_BRANCH_1_INPORT1[i];
```

El parámetro PER indica una probabilidad de que el paquete que atraviesa por dicha conexión sea corrompido y llegue con errores a su destino. Por esta razón, el parámetro PER puede usarse para modelar ruido aleatorio o, interrupciones en el canal: un PER=1 indica que los paquetes que atravesasen dicho canal siempre van a contener errores, de manera que esto se puede aprovechar en el módulo receptor para evaluar la calidad de los paquetes y descartar aquellos que estén corrompidos, simulando así la pérdida de dichos paquetes por caída del enlace.

Para evaluar el desempeño de la arquitectura frente a pérdidas de enlaces, se ejecutaron rondas de simulación variando el parámetro PER en el enlace entre las *Merging Units* de la rama 1 y el *switch* 1. En cada ronda, se probó con una *Merging Units* diferente y se tomaron tres valores para el parámetro PER: 0.1, correspondiente a un canal con alto ruido; 0.5, para simular un canal intermitente o demasiado ruidoso; 1 para simular un canal inexistente o la pérdida del enlace.

En todos los casos se desactivó el filtrado VLAN en ambos *switches*. La Figura 9 - 15 muestra la ubicación del enlace defectuoso para el caso de la *Merging Units* genérica.

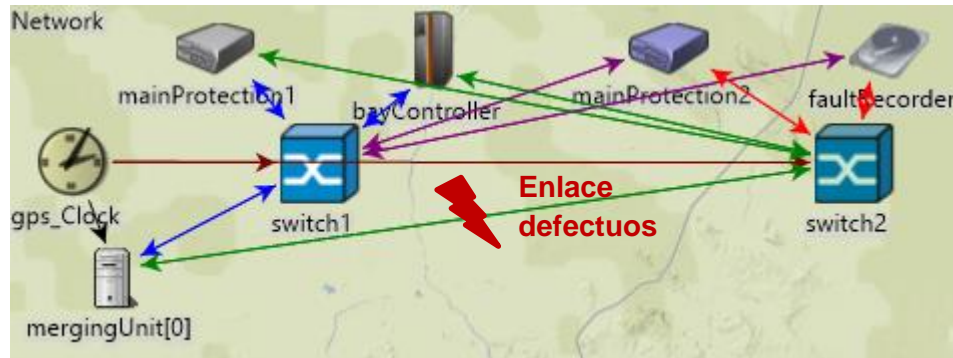


Figura 9 - 15: Representación gráfica del enlace defectuoso (elaboración propia)

La Tabla 9 - 9 muestra los resultados para la *Merging Units* genérica, la Tabla 9 - 10 para la Reason MU320, la Tabla 9 - 11 para las *Merging Units* SAM600CT y SA600VT (que de nuevo produjeron los mismos resultados), la Tabla 9 - 12 para la SIPROTEC 6MU805 y la Tabla 9 - 13 para la AMU VIZIMAX.

Tabla 9 - 9: Pérdida de enlace con la MU genérica (elaboración propia)

PER	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida R1	Tasa Pérdida R2
0.1	4801	4801	4800	4323	1.22000E-04	0.0208%	9.9563%
0.5	4801	4801	4800	2393	1.33000E-04	0.0208%	50.1562%
1	4801	4801	4799	1	1.60000E-04	0.0417%	99.9792%

Tabla 9 - 10: Pérdida de enlace con la Reason MU320 (elaboración propia)

PER	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida R1	Tasa Pérdida R2
0.1	4801	4801	4780	4313	4.34500E-03	0.4374%	10.1645%
0.5	4801	4801	4780	2377	4.33996E-03	0.4374%	50.4895%
1	4801	4801	4780	0	4.32800E-03	0.4374%	100.0000%

Tabla 9 - 11: Pérdida de enlace con la SAM600CT y SAM600VT (elaboración propia)

PER	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida R1	Tasa Pérdida R2
0.1	4801	4801	4791	4317	2.14500E-03	0.2083%	10.0812%
0.5	4801	4801	4791	2395	2.14000E-03	0.2083%	50.1146%
1	4801	4801	4791	0	2.12800E-03	0.2083%	100.0000%

Tabla 9 - 12: Pérdida de enlace con la SIPROTEC 6MU805 (elaboración propia)

PER	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida R1	Tasa Pérdida R2
0.1	4801	4801	4773	4306	5.74500E-03	0.5832%	10.3104%
0.5	4801	4801	4773	2372	5.73995E-03	0.5832%	50.5936%
1	4801	4801	4773	0	5.72800E-03	0.5832%	100.0000%

Tabla 9 - 13: Pérdida de enlace con la AMU VIZIMAX (elaboración propia)

PER	Envíos R1	Envíos R2	Recepciones	Duplicados	Retardo	Tasa Pérdida R1	Tasa Pérdida R2
0.1	4801	4801	4800	4333	2.45080E-04	0.0208%	9.7480%
0.5	4801	4801	4800	2389	2.23996E-04	0.0208%	50.2395%
1	4801	4801	4800	0	2.28000E-04	0.0208%	100.0000%

Desde la Tabla 9 - 9 hasta la Tabla 9 - 13 se puede apreciar que se evaluaron dos tasas de pérdida independientes: Tasa Pérdida R1 = Recepciones/Envíos R1; y Tasa Pérdida R2 = Duplicados/Envíos R2. Esta es la misma ecuación de (Aranda, 2014) aplicada independientemente sobre los envíos y recepciones de cada rama, para poder comparar el efecto de un enlace defectuoso en una de las ramas contra un enlace bueno en la otra. Para todas las *Merging Units* se observa un comportamiento similar: mientras la “Tasa Pérdida R1” se mantiene constante, la “Tasa Pérdida R2” oscila entre valores cercanos al 10% y hasta el 100% al variar el parámetro PER. Esto valida el funcionamiento de la arquitectura y la redundancia PRP: al llevar PER a un valor de 1 en uno de los enlaces, todos los paquetes que pasan por dicho enlace se pierden, mientras que los que pasan por el otro enlace permanecen intactos.

Por otro lado, todas las *Merging Units* produjeron menores tiempos de retardo cuando se presentó la pérdida de uno de los enlaces, a excepción de la *Merging Units* genérica. Esto puede interpretarse como una descongestión en los canales de comunicaciones tras la pérdida del enlace entre la *Merging Units* y el *switch*, debido a que llegan menos paquetes a dicho *switch* y se replican menos paquetes hacia los demás dispositivos.

Análisis de la sensibilidad de la topología ante pérdida de enlace o cambios en los niveles de ruido

Una vez más, es interesante analizar los cambios en los valores de la Tabla 9 - 9 hasta la Tabla 9 - 13, para identificar cuáles dispositivos producen mayor sensibilidad en la topología del bus de proceso, ante la pérdida de enlaces.

Por ejemplo, para la Reason MU320 de la Tabla 9 - 10, el retardo para un *PER* de 0,1 fue de 4.345ms, mientras que para un *PER* de 1 fue de 4.328ms. Esto representa una disminución de un 0.39% en el tiempo de retardo ante la pérdida del enlace.

Para las *Merging Units* SAM 600CT y SAM 600VT la disminución del retardo fue del 0.79%; para la SIPROTEC 6MU805 la reducción en el retardo es del 0.3%; y para la AMU VIZIMAX es del 6.97%. Se puede concluir entonces que la AU VIZIMAX es la *Merging Units* más sensible a pérdidas de enlace, en cuanto a variaciones en el tiempo de retardo.

También es interesante analizar la sensibilidad de las *Merging Units* ante canales ruidosos, lo cual se ve reflejado en variaciones de la tasa de pérdida. Por ejemplo, para la Reason MU320 de la Tabla 9 - 10, se tenía una “Tasa Perdida R2” de 10.1645% cuando el *PER* era de 0.1, mientras que la “Tasa Perdida R2” fue 50.4895% cuando el *PER* ascendió a 0.5. De esta forma, se puede interpretar que la topología del bus de proceso implementada con esta *Merging Units* experimentó un incremento del 40.3249% en la tasa de pérdida de paquetes ante un incremento de 0.1 a 0.5 en el ruido del canal.

Siguiendo el mismo razonamiento, se puede concluir que para las *Merging Units* SAM 600CT y SAM 600VT la variación ante el ruido fue del 40.033%, para la SIPROTEC 6MU805 fue del 40.2833% y para la AMU VIZIMAX fue del 40.4916%.

10. Validación de resultados obtenidos

Para la validación de la funcionalidad del ambiente de simulación desarrollado, se realizaron pruebas con dispositivos físicos reales en el Laboratorio IEC 61850 de la Universidad Nacional de Colombia – Sede Medellín. Las pruebas consistieron en implementar la arquitectura PRP con dos *Merging Units* distintas: la Reason MU320 y la SIPROTEC 6MU805. El tráfico generado por estos dispositivos fue capturado mediante el aplicativo Wireshark y comparado con el tráfico generado por el ambiente de simulación en OMNET++.

10.1 Validación del tráfico con la *Merging Units* Reason MU320

Se implementó una arquitectura como la de la Figura 9 - 1, utilizando dos *Merging Units* Reason MU320. El tráfico capturado sobre la LAN A se muestra en la Figura 10 - 1 y el tráfico sobre la LAN B se puede observar en la Figura 10 - 2. En estas figuras, la columna “Time” contiene el tiempo de llegada de los paquetes, “Source” muestra que provienen de la ReasonMU320, “Protocol” evidencia que se está trabajando bajo IEC 61850 y “Length” indica que la longitud de los paquetes *Sampled Value* es 128 bytes.

Para mayor detalle sobre el contenido de los paquetes *Sampled Value* capturados por Wireshark, se puede revisar la Figura 10 - 3.

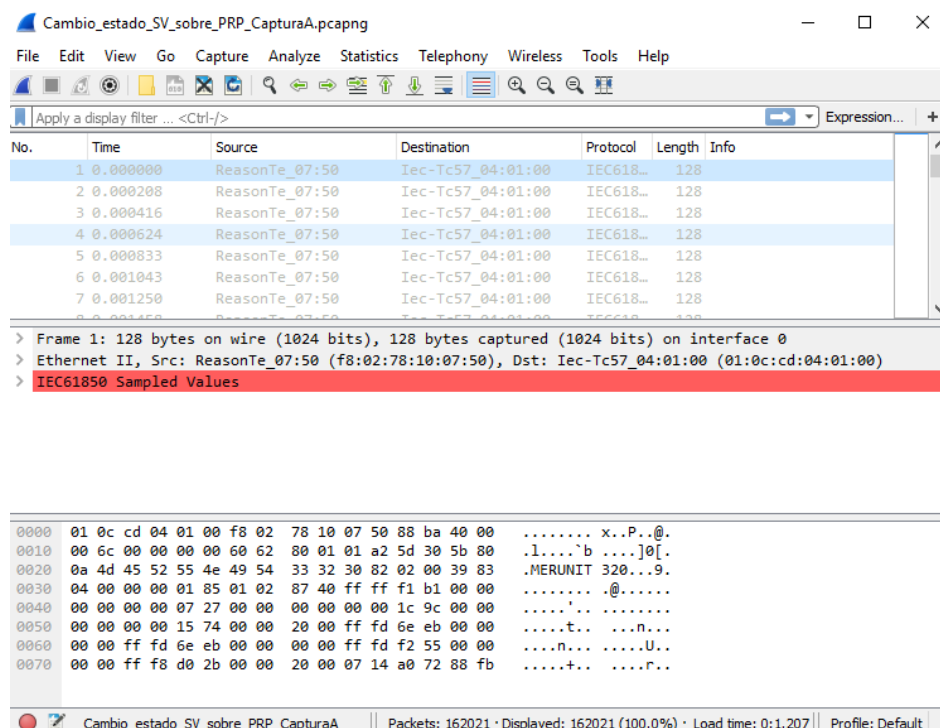


Figura 10 - 1: Captura del tráfico sobre la LAN A – Reason MU320 (suministrada por Laboratorio IEC 61850 UNAL)

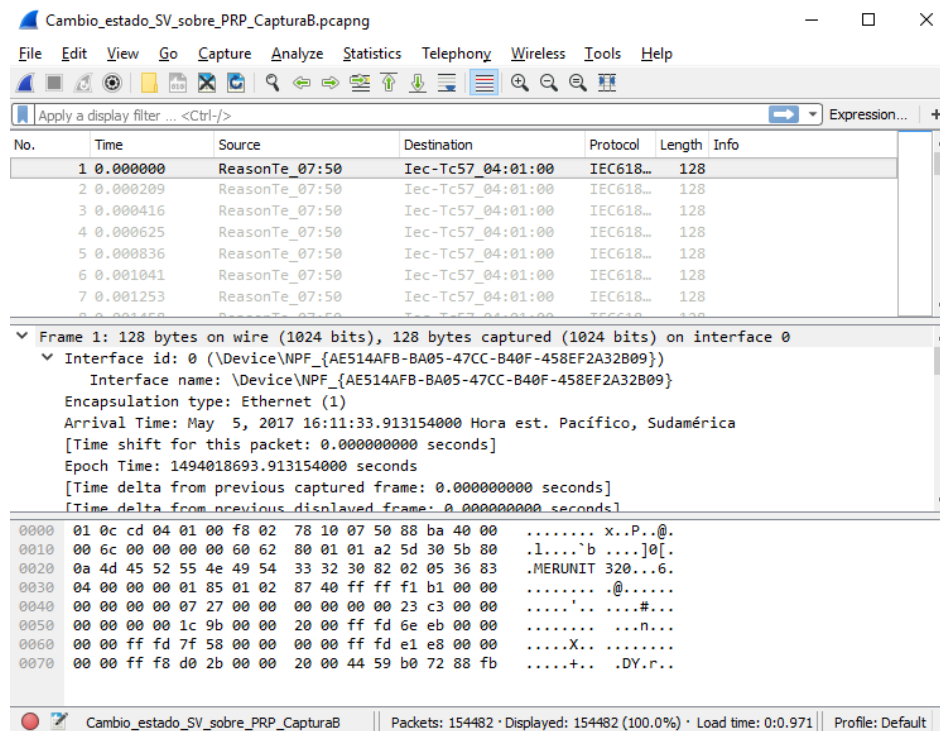


Figura 10 - 2: Captura de tráfico sobre LAN B – Reason MU320 (suministrada por Laboratorio IEC 61850 UNAL)

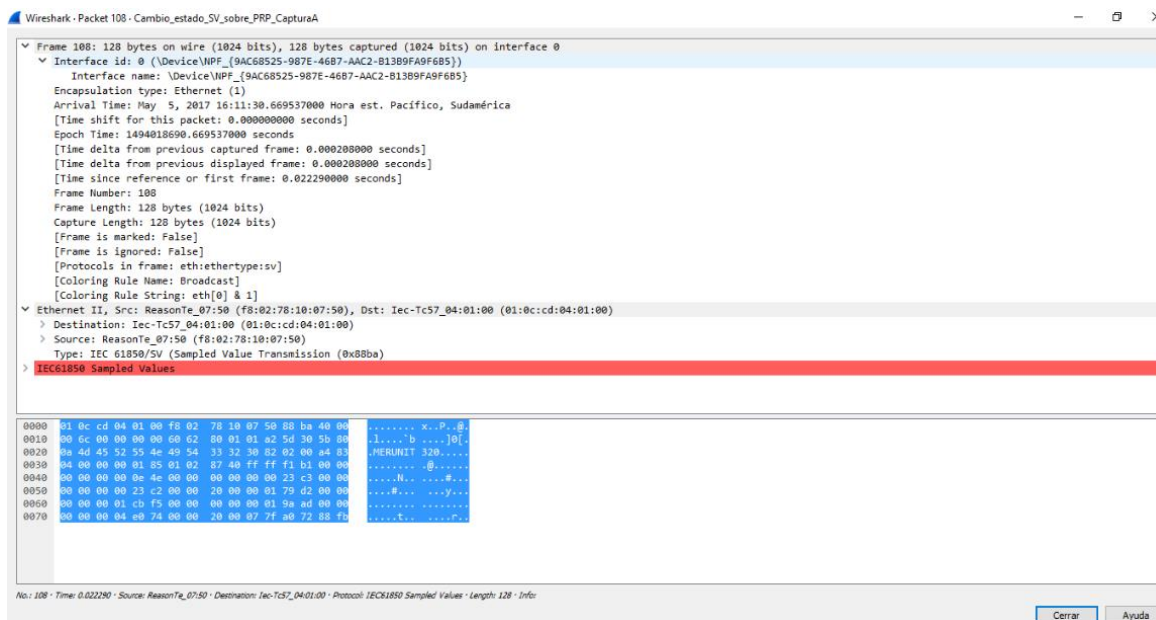


Figura 10 - 3: Detalle del contenido del paquete *Sampled Value* (suministrada por Laboratorio IEC 61850 UNAL)

A partir de los datos de la columna “Time” en la Figura 10 - 1 y la Figura 10 - 2 se construyó la Tabla 10 - 1. La columna “Tasa de muestreo” se construye como la diferencia entre los tiempos de llegada de dos paquetes consecutivos (por ejemplo, $416-208=208$). Por esta razón, se puede considerar que esta columna hace referencia al espaciamiento temporal entre dos paquetes consecutivos y puede ser entendida como la tasa de muestreo.

La Tabla 10 - 1 permite evidenciar que en el ambiente de laboratorio se obtiene una tasa de muestreo muy cercana al ideal del estándar IEC 61850, la cual es de 80 muestras por ciclo o una muestra cada 208.33µs.

Tabla 10 - 1: Tasa de muestreo de la Reason MU320

LAN A		LAN B	
Tiempo de llegada	Tasa de muestreo	Tiempo de llegada	Tasa de muestreo
0		0	
208	208	209	209
416	208	416	207
624	208	625	209
833	209	836	211
1043	210	1041	205
1250	207	1253	212
Promedio	208.333333	Promedio	208.833333

Esta misma arquitectura fue implementada en OMNET++, con el fin de determinar si es posible obtener resultados comparables, como se aprecia en la Figura 10 - 4, donde el círculo rojo muestra el contenido de un paquete y el círculo naranja muestra el tráfico capturado.

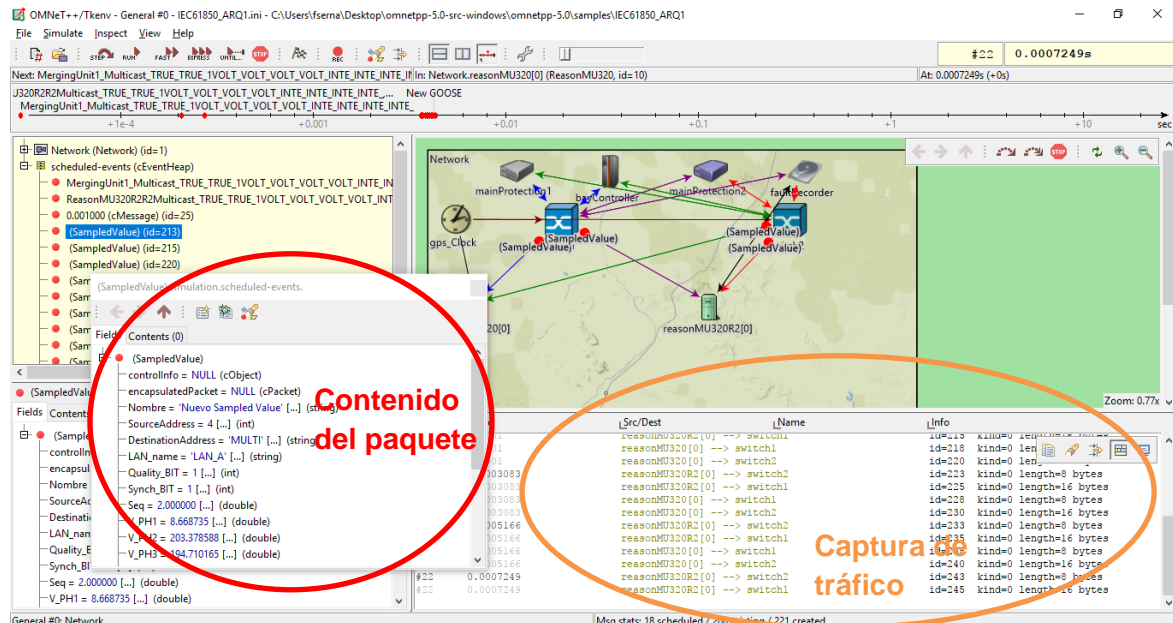


Figura 10 - 4: Simulación de captura de tráfico en OMNET++ para la Reason MU320 (elaboración propia)

Los detalles del contenido de cada paquete se muestran en la Figura 10 - 5. En comparación con la Figura 10 - 3 se puede ver que OMNET++ no presenta información del encapsulamiento del paquete en tramas Ethernet, como sí lo hace Wireshark; sin embargo, OMNET++ sí permite llegar a un nivel de detalle mayor sobre el contenido del paquete, con mucha más facilidad para identificar los bits de calidad, sincronización y la tensión en cada una de las fases del *Sampled Value*.

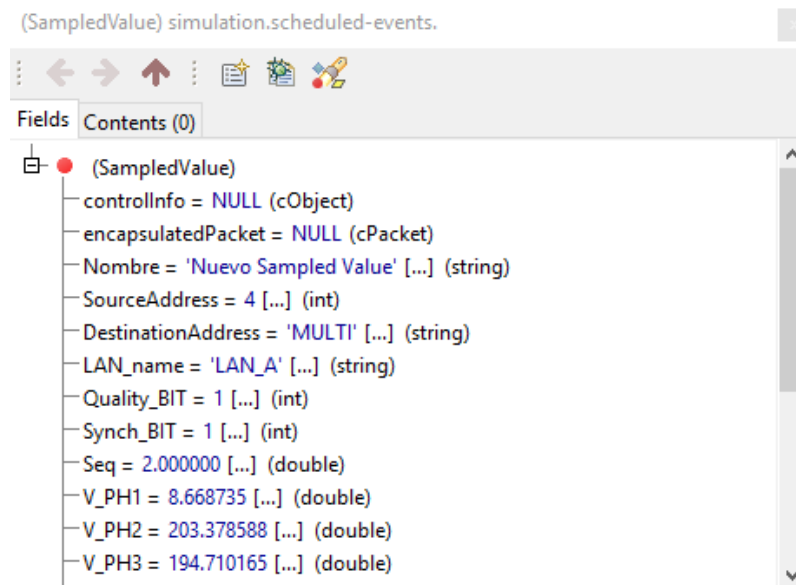


Figura 10 - 5: Contenido del paquete *Sampled Value* en OMNET++ (elaboración propia)

Por otro lado, el detalle del tráfico capturado por OMNET++ se muestra en Figura 10 - 6. En comparación con la Figura 10 - 1 y la Figura 10 - 2, se puede concluir que OMNET++ mezcla el tráfico de las LAN A y B, a diferencia de Wireshark. Esto se puede reconocer con la dirección destino de los *Sampled Values*, que alterna entre *switch 1* y *switch 2* para paquetes consecutivos.

De forma análoga a la Tabla 10 - 1, la Tabla 10 - 2 fue construida a partir de los datos de la columna "Time" de la Figura 10 - 6 y restando los tiempos de llegada entre dos paquetes consecutivos para formar la columna "Tasa de muestreo". Se puede concluir que los tiempos de muestreo de la Reason MU320 obtenidos con OMNET++ son equivalentes a los obtenidos en el Laboratorio IEC 61850.

Event#	Time	Src/Dest	Name	Info
#18	0.0003083	reasonMU320R2[0] --> switch2		id=223 kind=0 len=8 bytes
#18	0.0003083	reasonMU320R2[0] --> switch1		id=225 kind=0 len=8 bytes
#19	0.0003083	reasonMU320[0] --> switch1		id=228 kind=0 length=8 bytes
#19	0.0003083	reasonMU320[0] --> switch2		id=230 kind=0 length=16 bytes
#20	0.0005166	reasonMU320R2[0] --> switch2		id=233 kind=0 length=8 bytes
#20	0.0005166	reasonMU320R2[0] --> switch1		id=235 kind=0 length=16 bytes
#21	0.0005166	reasonMU320[0] --> switch1		id=238 kind=0 length=8 bytes
#21	0.0005166	reasonMU320[0] --> switch2		id=240 kind=0 length=16 bytes
#22	0.0007249	reasonMU320R2[0] --> switch2		id=243 kind=0 length=8 bytes
#22	0.0007249	reasonMU320R2[0] --> switch1		id=245 kind=0 length=16 bytes
#23	0.0007249	reasonMU320[0] --> switch1		id=248 kind=0 length=8 bytes
#23	0.0007249	reasonMU320[0] --> switch2		id=250 kind=0 length=16 bytes
#24	0.0009332	reasonMU320R2[0] --> switch2		id=253 kind=0 length=8 bytes
#24	0.0009332	reasonMU320R2[0] --> switch1		id=255 kind=0 length=16 bytes
#25	0.0009332	reasonMU320[0] --> switch1		id=258 kind=0 length=8 bytes
#25	0.0009332	reasonMU320[0] --> switch2		id=260 kind=0 length=16 bytes
#26	0.001	gps_Clock --> reasonMU320[0] 0.001000		id=25 kind=0
#28	0.0011415	reasonMU320R2[0] --> switch2		id=265 kind=0 length=8 bytes
#28	0.0011415	reasonMU320R2[0] --> switch1		id=267 kind=0 length=16 bytes
#29	0.0011415	reasonMU320[0] --> switch1		id=270 kind=0 length=8 bytes

Figura 10 - 6: Tráfico capturado por OMNET++ para la Reason MU320 (elaboración propia)

Tabla 10 - 2: Tasa de muestreo según OMNET++

LAN A		LAN B	
Tiempo de llegada	Tasa de muestreo	Tiempo de llegada	Tasa de muestreo
308.3		308.3	
516.6	208.3	516.6	208.3
724.9	208.3	724.9	208.3
933.2	208.3	933.2	208.3
1141.5	208.3	1141.5	208.3
1349.8	208.3	1349.8	208.3
1558.1	208.3	1558.1	208.3
Promedio	208.3	Promedio	208.3

10.2 Validación del tráfico con la *Merging Units* SIPROTEC 6MU805

De forma análoga al caso anterior, se implementó la arquitectura de la Figura 9 - 1, esta vez empleando dos *Merging Units* SIPROTEC 6MU805. La Figura 10 - 7 muestra la captura del tráfico sobre la LAN A y la Figura 10 - 8 sobre la LAN B.

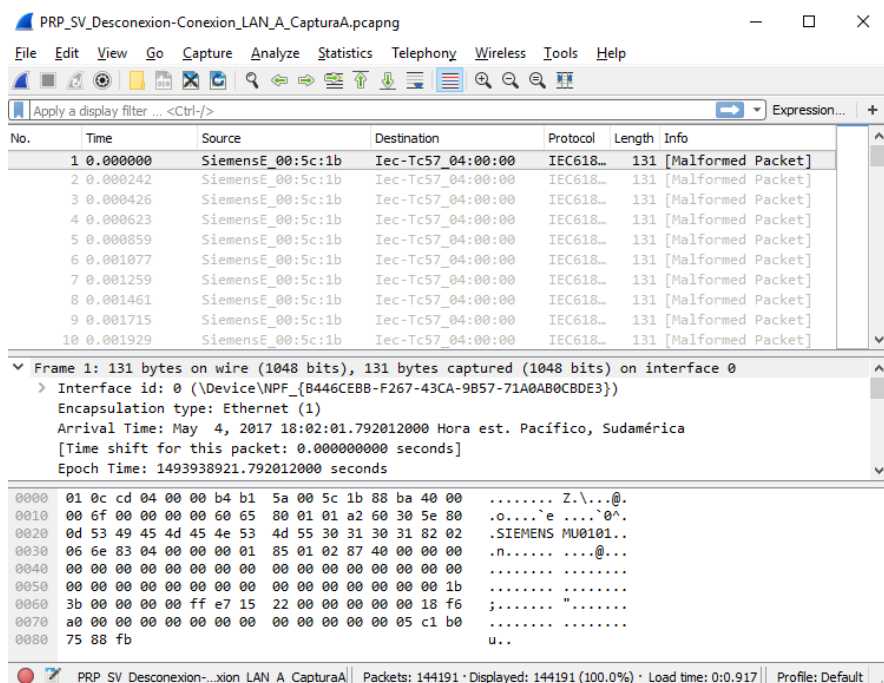


Figura 10 - 7: Captura de tráfico LAN A – SIPROTEC 6MU850 (suministrada por Laboratorio IEC 61850 UNAL)

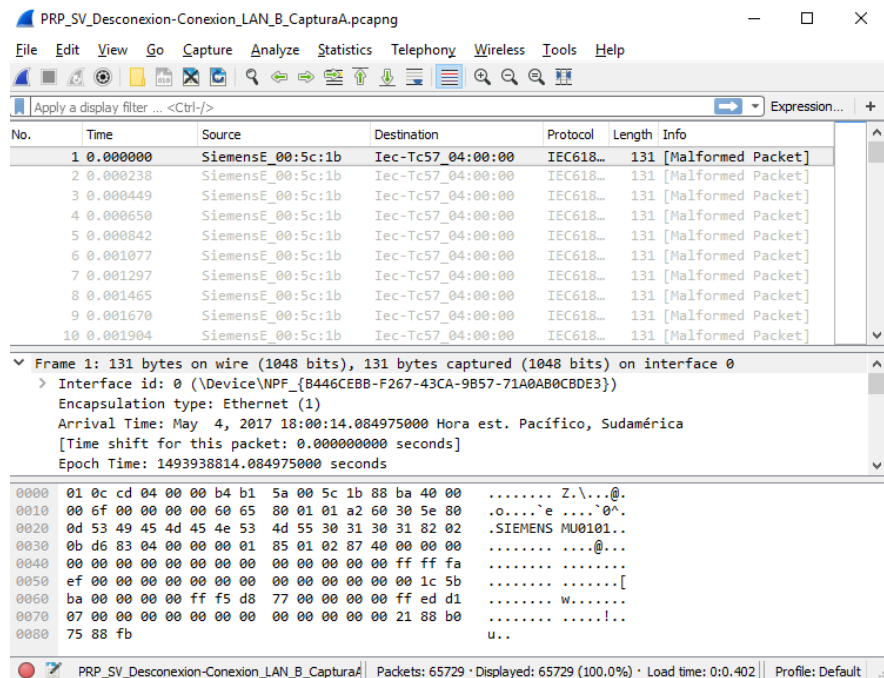


Figura 10 - 8: Captura de tráfico LAN B – SIPROTEC 6MU850 (suministrada por Laboratorio IEC 61850 UNAL)

Para este caso, se obtiene un tiempo de llegada promedio entre paquetes de 209.83us sobre la LAN A y de 216.17us sobre la LAN B, como se observa en la Tabla 10 - 3. En este caso el tiempo de muestreo sobre la LAN A se desvía 1.5us del tiempo ideal de muestreo (208.33us) y la LAN B se desvía 7.84us, indicando que esta *Merging Units* puede tener tiempos de procesamiento un poco mayores que los de la Reason MU320.

Tabla 10 - 3: Tasa de muestreo para la SIPROTEC 6MU805

LAN A		LAN B	
Tiempo de llegada	Tasa de muestreo	Tiempo de llegada	Tasa de muestreo
0		0	
242	242	238	238
426	184	449	211
623	197	650	201
859	236	842	192
1077	218	1077	235
1259	182	1297	220
Promedio	209.8333333	Promedio	216.1666667

Nuevamente, para verificar la funcionalidad del ambiente de simulación, se implementó la misma arquitectura en OMNET++. La Figura 10 - 9 muestra la captura de tráfico obtenida con el simulador. La Tabla 10 - 4 presenta las tasas de muestreo obtenidas a partir de la columna "Time". En este caso, se observa que OMNET++ permite obtener resultados más cercanos al ideal de muestreo, mientras que Wireshark permitió capturar ligeros desvíos de dichos tiempos. Esto se debe a que el simulador trabaja con parámetros fijos programados desde el comienzo de la simulación, mientras que el laboratorio está expuesto a irregularidades reales de circuitos operativos.

Este último punto es importante porque evidencia las limitaciones de la herramienta de simulación: una vez seleccionados unos parámetros, las simulaciones del mismo sistema en las mismas condiciones mostrarán resultados muy similares entre sí, ignorando otras variables como la temperatura del medio, el ruido de fondo, los defectos en las conexiones, entre muchos otros que sí pueden ser puestos a prueba en el ambiente de laboratorio.

El ambiente de simulación debería entonces emplearse para identificar cuáles topologías son más prometedoras en determinadas situaciones, para luego validar su funcionamiento real en el laboratorio. Por ejemplo, con el ambiente de simulación sería posible determinar la cantidad de *Merging Units* adecuada para una subestación en configuración estrella, que permita cumplir con los requisitos de calidad deseados (como un retardo máximo permitido); una vez identificado el número de *Merging Units*, la topología real debería implementarse en el laboratorio para obtener mayor precisión en los índices de desempeño del diseño propuesto.

Event#	Time	Src/Dest	Name	Info
#17	0.0001	sIPROTEC_6MU805[0] --> switch1		id=218 kind=0 len=
#17	0.0001	sIPROTEC_6MU805[0] --> switch2		id=220 kind=0 len=
#18	0.0003083	sIPROTEC_6MU805_R2[0] --> switch2		id=223 kind=0 length=8 bytes
#18	0.0003083	sIPROTEC_6MU805_R2[0] --> switch1		id=225 kind=0 length=16 bytes
#19	0.0003083	sIPROTEC_6MU805[0] --> switch1		id=228 kind=0 length=8 bytes
#19	0.0003083	sIPROTEC_6MU805[0] --> switch2		id=230 kind=0 length=16 bytes
#20	0.0005166	sIPROTEC_6MU805_R2[0] --> switch2		id=233 kind=0 length=8 bytes
#20	0.0005166	sIPROTEC_6MU805_R2[0] --> switch1		id=235 kind=0 length=16 bytes
#21	0.0005166	sIPROTEC_6MU805[0] --> switch1		id=238 kind=0 length=8 bytes
#21	0.0005166	sIPROTEC_6MU805[0] --> switch2		id=240 kind=0 length=16 bytes
#22	0.0007249	sIPROTEC_6MU805_R2[0] --> switch2		id=243 kind=0 length=8 bytes
#22	0.0007249	sIPROTEC_6MU805_R2[0] --> switch1		id=245 kind=0 length=16 bytes
#23	0.0007249	sIPROTEC_6MU805[0] --> switch1		id=248 kind=0 length=8 bytes
#23	0.0007249	sIPROTEC_6MU805[0] --> switch2		id=250 kind=0 length=16 bytes
#24	0.0009332	sIPROTEC_6MU805_R2[0] --> switch2		id=253 kind=0 length=8 bytes
#24	0.0009332	sIPROTEC_6MU805_R2[0] --> switch1		id=255 kind=0 length=16 bytes
#25	0.0009332	sIPROTEC_6MU805[0] --> switch1		id=258 kind=0 length=8 bytes
#25	0.0009332	sIPROTEC_6MU805[0] --> switch2		id=260 kind=0 length=16 bytes
#26	0.001	gps_Clock --> sIPROTEC_6MU805[0] 0.001000		id=25 kind=0
#28	0.0011415	sIPROTEC_6MU805_R2[0] --> switch2		id=265 kind=0 length=8 bytes
#28	0.0011415	sIPROTEC_6MU805_R2[0] --> switch1		id=267 kind=0 length=16 bytes
#29	0.0011415	sIPROTEC_6MU805[0] --> switch1		id=270 kind=0 length=8 bytes
#29	0.0011415	sIPROTEC_6MU805[0] --> switch2		id=272 kind=0 length=16 bytes

Figura 10 - 9: Captura de tráfico en OMNET++ para la SIPROTEC 6MU805

Tabla 10 - 4: Tasa de muestreo según OMNET++ para la SIPROTEC 6MU805

LAN A		LAN B	
Tiempo de Llegada	Retardo	Tiempo de Llegada	Retardo
100		100	
308.3	208.3	308.3	208.3
516.6	208.3	516.6	208.3
724.9	208.3	724.9	208.3
933.2	208.3	933.2	208.3
1141.5	208.3	1141.5	208.3
1349.8	208.3	1349.8	208.3
Promedio	208.3	Promedio	208.3

11. Conclusiones y recomendaciones

11.1 Conclusiones

La revisión del estado del arte permitió identificar un buen volumen de investigación en temas como la realización del bus de proceso con diversas topologías, los métodos para seleccionarlasy la simulación y evaluación del desempeño de determinadas arquitecturas. Algunos autores ya han empleado herramientas similares a OMNET++ para modelar el bus de proceso. Sin embargo, los trabajos encontrados se concentran en sólo en algunos de los servicios de la norma IEC 61850, o se basan en índices muy MTTR y MTTF muy generales para determinar la confiabilidad de la red. El presente trabajo va un poco más allá, basando los modelos en información técnica de proveedores reales, simulando además varios servicios IEC 61850 como *Sampled Values*, GOOSE o PTP. El ambiente de simulación desarrollado cuenta pues con un enfoque más práctico, más orientado a la aplicación en condiciones reales para Colombia.

La estimación de los índices de desempeño para dispositivos comerciales se realizó mediante la consulta de catálogos y fichas técnicas de fabricantes, y a partir de correspondencia intercambiada con los mismos. A partir de ello se pudo evidenciar que los índices *Mean Time To Failure – MTTF* y *Mean Time To Repair – MTTR*, inicialmente supuestos como relevantes para la evaluación del desempeño del bus de proceso, no resultan prácticos a la hora de implementar un ambiente de simulación orientado al tráfico de comunicaciones. Esto se debe a dos razones: en primer lugar, la información disponible sobre dispositivos comerciales difícilmente especifica los parámetros MTTF y MTTR, de manera que no es posible incorporar estos datos en los modelos de simulación de la mayoría de dispositivos; en segundo lugar, el índice MTTF puede llegar a tener valores del orden de años, así que es más adecuado para medir la confiabilidad del sistema que su desempeño temporal.

La caracterización de los dispositivos comerciales también mostró que los parámetros relevantes para la evaluación del desempeño del bus de proceso, contenidos en la información suministrada por fabricantes, son los tiempos de procesamiento, la velocidad de transmisión y el número de puertos de cada dispositivo comercial. Sin embargo, esta información no se encuentra homologada o estandarizada para diferentes fabricantes, pues no en todos los catálogos se encuentran referencias claras a los tiempos de transmisión o retardo.

Es posible entonces construir modelos de cada dispositivo a partir de la información contenida en catálogos y fichas técnicas, pero se recomienda validar el modelo con pruebas de laboratorio, que disminuyan la incertidumbre sobre la información ambigua o poco homologada que suministran los fabricantes.

La caracterización de requerimientos del bus de proceso para las subestaciones del SIN colombiano se obtuvo a partir de una muestra de unifilares publicados por la UPME y de las recomendaciones de la literatura sobre la asignación de equipos a cada bahía. Esta caracterización mostró que la mayor parte de las subestaciones presentan una configuración de interruptor y medio o doble barra más acople de transferencia, lo cual facilita la aplicación de arquitecturas estándar de bus de proceso.

En cuanto a los requisitos del bus de proceso establecidos por la UPME, se pudo constatar que éstos son muy generales y con frecuencia remiten a otras normas como la IEC 60186, IEC 60870 o IEEE C37.111. De igual manera, la resolución CREG 025 de 1995 establece requisitos muy generales de disponibilidad y confiabilidad del sistema, sin ahondar en detalles del desempeño temporal de las redes de comunicación. En conclusión, la regulación vigente no es suficientemente explícita como para ser considerada en la selección de una arquitectura de bus de proceso para una subestación. Por esta razón, y de acuerdo con las indicaciones de la literatura y de la norma IEC 61850, se recomienda una arquitectura estándar para las subestaciones del SIN colombiano, que cuente con redundancia PRP o HSR y que sea replicable en cada bahía de la subestación.

Si bien la regulación vigente en Colombia no brinda elementos para seleccionar una topología de bus de proceso en una subestación, sí podría utilizarse en un ambiente de simulación para establecer los parámetros de desempeño mínimos por debajo de los cuales los resultados de la red serían inaceptables.

La plataforma OMNET++ permitió integrar la información recopilada sobre los dispositivos comerciales para el bus de proceso y las arquitecturas recomendadas para el SIN colombiano. Con esta herramienta fueron programados los modelos de las *Merging Units* de cuatro fabricantes reales, la configuración dinámica de una arquitectura PRP y la captura de información de simulaciones para evaluar el desempeño del bus de proceso.

Si bien, al comienzo de este trabajo se había planteado que la aplicación de simulación debería permitir comparar el desempeño de arquitecturas de bus de proceso en términos de confiabilidad, disponibilidad y costos, la falta de disponibilidad de información o la ambigüedad en los índices MTTF y MTTR condujeron la investigación hacia un enfoque más práctico de la evaluación del desempeño del bus de proceso, en términos de retardos y pérdidas de paquetes. Para este último enfoque, la herramienta OMNET++ resulta adecuada.

El ambiente de simulación desarrollado en OMNET++ permitió evaluar el desempeño de las arquitecturas de bus de proceso a partir de diferentes variables: el retardo en el envío de paquetes, la tasa de pérdida de paquetes, el número de recepciones y duplicados en la redundancia PRP, así como la sensibilidad de la red ante cambios en un dispositivo con respecto a las anteriores variables.

Las validaciones realizadas con el apoyo del Laboratorio IEC 61850 de la Universidad Nacional de Colombia – Sede Medellín evidenciaron que las capturas de tráfico generadas por el ambiente de simulación desarrollado se asemejan a las obtenidas en el laboratorio. En general, el ambiente de simulación puede recrear los resultados obtenidos con los montajes en el laboratorio. Sin embargo, los resultados del laboratorio pueden tener inicialmente mayor precisión que los del ambiente de simulación, debido a que los parámetros de los modelos en OMNET++ fueron tomados de información suministrada por fabricantes, la cual no siempre está homologada o verificada. Por esta razón, un buen ejercicio consistiría en parametrizar los modelos del ambiente de simulación con base en los resultados del laboratorio, a manera de calibración.

Por otra parte, de la comparación entre los resultados obtenidos con el ambiente de simulación y el Laboratorio IEC 61850, se puede concluir que la evaluación del desempeño del bus de proceso es más cómoda en el ambiente de simulación que en el laboratorio. La visualización de variables como la tasa de pérdida de paquetes o el retardo punto a punto, es muy fácilmente verificable en el ambiente de simulación, en comparación con la disposición de los datos que aparecen en las capturas Wireshark del laboratorio. Por esta razón, se considera que el ambiente de simulación y el Laboratorio IEC 61850 pueden llegar a ser un buen complemento a la hora de modelar y evaluar el desempeño de topologías de bus de proceso para subestaciones colombianas.

11.2 Recomendaciones

Debido a la falta de homologación de la información técnica en catálogos de diversos fabricantes, se recomienda que los modelos que en un futuro se implementen sobre el ambiente desarrollado sean siempre validados en un laboratorio como el Laboratorio IEC 61850 de la Universidad Nacional. Variables como el tiempo de procesamiento pueden ser medidas con mayor confiabilidad en estos laboratorios, para ser luego ingresadas a los modelos de simulación.

Trabajos futuros podrían orientarse a la ampliación de la base de datos de dispositivos comerciales, de manera que se cuente con un catálogo más amplio de modelos para simular. Los modelos podrían extenderse para que fueran más realistas, configurando por ejemplo las tramas de los paquetes con un nivel de detalle mayor. Otras topologías de bus de proceso podrían ser implementadas también empleando OMNET++.

Se recomienda integrar, de forma complementaria, el ambiente de simulación desarrollado con el Laboratorio IEC 61850, ya que algunos resultados de evaluación de desempeño pueden visualizarse más fácilmente en el ambiente de simulación, mientras que el laboratorio puede proveer insumos para crear modelos más precisos de cada dispositivo.

12. Referencias bibliográficas

ABB. (2007). Bay control IED REC 670. Retrieved from

https://library.e.abb.com/public/6fd8bdb6db6394d8c125729000765648/1MRK511154-BEN_G_en_Bay_control_IED_REC670_open_configuration.pdf

ABB. (2011). 615 series ANSI. Retrieved from

https://library.e.abb.com/public/552f051b7da9d407c12578f800288693/RE_615ANSI_tech_050144_ENc.pdf

ABB. (2012). Non conventional instrument transformers. Retrieved from

[http://www02.abb.com/global/gad/gad02181.nsf/0/972b57391b2140d0c1257a620046043b/\\$file/Non-conventional+instrument+transformers+-+Advanced+GIS+substations+with+IEC+61850-9-2+LE+process+bus.pdf](http://www02.abb.com/global/gad/gad02181.nsf/0/972b57391b2140d0c1257a620046043b/$file/Non-conventional+instrument+transformers+-+Advanced+GIS+substations+with+IEC+61850-9-2+LE+process+bus.pdf)

ABB. (2013). Protection and Control IED Manager PCM600. Retrieved from

[http://www04.abb.com/global/seitp/seitp202.nsf/0/d13cebbbc495afa9bc1257b1e0028edfa/\\$file/PCM600_pg_756448_ENh.pdf](http://www04.abb.com/global/seitp/seitp202.nsf/0/d13cebbbc495afa9bc1257b1e0028edfa/$file/PCM600_pg_756448_ENh.pdf)

ABB. (2016). SAM600 Process Bus I/O System. Retrieved from

https://library.e.abb.com/public/f43c62927bf04498a264a1642fef2440/1MRK511428-BEN_-_en_Product_guide__Process_Bus_I_O_System_SAM600_version_1.1.pdf

Alstom. (2012a). MiCOM H Series. Retrieved from

<https://www.gegridsolutions.com/alstomenergy/grid/Global/Grid/Resources/Documents/Automation/SAS/MiCOM-H-Series-Brochure-GB-epslanguage=en-GB.pdf>

Alstom. (2012b). COSI-MU *Merging Units*.

Alstom. (2013a). MU-Agile XMU800.

Alstom. (2013b). MU Agile AMU Technical Manual. Retrieved from

<https://www.gegridsolutions.com/alstomenergy/grid/Global/Grid/Resources/Documents/Automation/Agile%20related/MU%20Agile%20AMU-TM-EN-1-epslanguage=en-GB.pdf>

Alstom. (2015). MiCOM C264 Modular substation controller. Retrieved from

http://www.ase.cz/sites/default/files/upload/files/MiCOM/C264_brochure.pdf

Apostolov, A., Auperrin, F., Passet, R., Guenego, M., & Guilles, F. (2006). A Distributed Recording System Based on IEC 61850 Process Bus. *Advanced Metering, Protection, Control, Communication, and Distributed Resources*.

Aranda, J. M. (2014). Diseño y evaluación del desempeño de una red de comunicaciones para medición inteligente en Network Simulator–2. *Revista Facultad de Ingeniería - Universidad Distrital Francisco José de Caldas*, 20(1).

Barron, D. A., & Holliday, P. (2010). Implementation of IEC 61850 Process Bus. A Utility View on Design, Installation, Testing and Commissioning, and Lifetime Issues. *10th IET International Conference on Developments in Power System Protection (DPSP 2010). Managing the Change*.

Chelluri, S., Dolezilek, D., Dearien, J., & Karla, A. (2014). Understanding and Validating Ethernet Networks for Mission-Critical Protection, Automation, and Control Applications. Retrieved from <https://www.selinc.com>.

Conti, M., & Giordano, S. (2014). Mobile ad hoc networking: milestones, challenges, and new research directions. *IEEE Communications Magazine*, 52(1), 85–96.

- CREG. (1995). Resolución número 025 del 13 de julio de 1995. Retrieved from [http://apolo.creg.gov.co/Publicac.nsf/2b8fb06f012cc9c245256b7b00789b0c/3a940408d14bf2e80525785a007a653b/\\$FILE/Cr025-95.pdf](http://apolo.creg.gov.co/Publicac.nsf/2b8fb06f012cc9c245256b7b00789b0c/3a940408d14bf2e80525785a007a653b/$FILE/Cr025-95.pdf)
- De Dominicis, C. M., Ferrari, P., Flammini, A., Rinaldi, S., & Quarantelli, M. (2010). Experimental Evaluation of Synchronization Solutions for Substation Automation Systems. *2010 IEEE Instrumentation & Measurement Technology Conference Proceedings*.
- Franco, J. H., García, J. D., Osorno, T., & Latozefski, C. (2015). Primera solución de bus de proceso para subestaciones de alta y extra alta tensión en Colombia. *Revista CIDET*.
- GE. (2014). MU320 *Merging Units* Analog and digital, fast and accurate. Retrieved from https://www.google.com.co/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&cad=rja&uact=8&ved=0ahUKEwjWslam6bvWAhXGRSYKHWYnDclQFggrMAE&url=https%3A%2F%2Fwww.gegridsolutions.com%2Fapp%2FDownloadFile.aspx%3Fprod%3DMU320%26type%3D1%26file%3D9&usg=AFQjCNE5w2aTALUks-SL_bpFDvwnGsLpzg
- GE. (2017). MU320 Technical Manual. Retrieved from http://www.gegridsolutions.com/alstomenergy/grid/Global/Grid/Resources/Documents/Automation/Reason%20products/MU320_TM_EN_6.pdf
- Giorgi, G., & Narduzzi, C. (2007). Modeling and Simulation Analysis of PTP Clock Servo. *2007 International IEEE Symposium on Precision Clock Synchronization (ISPCS)*.
- He, R., He, X., Zexiang, C., Xiaohua, L., & Lin, Z. (2014). Performance and reliability assessment of protection system in IEC 61850-based smart substation. *IEEE PES Innovative Smart Grid Technologies, Europe*.

- Hidalgo Quesada, J. D. (2008). Guía básica de diseño de subestaciones eléctricas con énfasis en el arreglo de barras colectora de interruptor y medio. Universidad de Costa Rica. Retrieved from http://eie.ucr.ac.cr/uploads/file/proybach/pb_08_II/pb0816t.pdf
- Hodder, S., Kaszteny, B., McGin, D., & Hunt, R. (2009). IEC 61850 Process Bus Solution Addressing Business Needs Of Today's Utilities. *Power Systems Conference, 2009. PSC '09*.
- IEC. (2009). IEC 61850-9-2: Communication networks and systems for power utility automation – Part 9-2: Specific Communication Service Mapping (SCSM) – *Sampled Values* over ISO/IEC 8802-3.
- IEC. (2010). IEC 61850-7-2.; Communication networks and systems for power utility automation – Basic information and communication structure – Abstract communication service interface (ASCI).
- IEC. (2011). IEC 61850-8-1.; Communication networks and systems for power utility automation – Specific communication service mapping (SCSM) – Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802- 3.
- IEC. (2013). IEC/TR 61850-90-4, Communication Networks and Systems for Power Utility Automation – Part 90-4: Network Engineering Guidelines.
- IEEE. (2008). IEEE Standard 1588-2008, IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems.
- INET Team. (2016). INET Framework for OMNeT++. Retrieved from <https://omnetpp.org/doc/inet/api-current/inet-manual-draft.pdf>
- Jiang, K., & Singh, C. (2010). Reliability Modeling of All-Digital Protection System Including Impact of Repair. *IEEE Transactions on Power Delivery*.

- Kasztenny, B., McGin, D., & Adamiak, M. (2009). An Optimized Architecture For IEC 61850 Process Bus. *Power Systems Conference, 2009*.
- Konka, J., Arthur, C., García, F., & Atkinson, R. (2011). Traffic Generation of IEC 61850 Sampled Values. *2011 IEEE First International Workshop on Smart Grid Modeling and Simulation (SGMS) - at IEEE SmartGridComm 2011*.
- Kumar, S., Das, N., & Islam, S. (2015). Performance Evaluation of a Process Bus Architecture in a Zone Substation based on IEC 61850-9-2. *Power and Energy Engineering Conference (APPEEC), 2015 IEEE PES Asia-Pacific*.
- Li, X., Li, H., & Wen, H. (2016). Evaluation of Data Flow Control Performance for Architecture of Secondary Substation System. *IEEE 8th International Power Electronics and Motion Control Conference (IPEMC-ECCE Asia)*.
- Liu, N., Panteli, M., & Crossley, P. A. (2014). Reliability evaluation of a substation automation system communication network based on IEC 61850. *12th IET International Conference on Developments in Power System Protection (DPSP 2014)*.
- Liu, X., Pang, J., Zhabg, L., & Xu, D. (2014). A High-Reliability and Determinacy Architecture for Smart Substation Process-Level Network Based on Cobweb Topology. *IEEE TRANSACTIONS ON POWER DELIVERY, 29(2)*.
- Mayorga, C., & Sarmiento, M. (2010). Modelo de evaluación financiera de subestaciones eléctricas tipo. Universidad Industrial de Santander. Retrieved from <https://es.scribd.com/document/283096152/Evaluacion-Financiera-de-S-E-At>
- Mike, H., & Drew, B. (2002). *Network+ Training Guide - Chapter 3: networking components and devices*. Pearson IT Certification. Retrieved from <http://catalogue.pearsoned.co.uk/samplechapter/0789732556.pdf>

- Mitalkumar, K. (2011). Investigating Performance and Reliability of Process Bus Networks for Digital Protective Relaying. *University of Western Ontario*.
- Mo, J., Tan, J. C., Crossley, P. A., Bo, Z. Q., & Klimek, A. (2010). Evaluation of process bus reliability. *10th IET International Conference on Developments in Power System Protection (DPSP 2010). Managing the Change*.
- Mohagheghi, S., Stoupis, J., & Wang, Z. (2009). Communication Protocols and Networks for Power Systems- Current Status and Future Trends.
- Noorhayati, M. N., Norashidah, M. D., Ejaz, A., & Awang, N. (2016). OMNET++ based cognitive radio simulation network. *2016 IEEE 7th Control and System Graduate Research Colloquium (ICSGRC 2016)*.
- Pesente, J., Rolim, J., Ramos, R., & Otto, R. (2016). Determination of Disturbance Patterns using Fault Recorder Data and Regression Trees. *2016 IEEE Power and Energy Society General Meeting (PESGM)*.
- Rodríguez, A. (2013). Sistema de Transmisión Nacional Eléctrico Colombiano: Nuevas Obras y Plan de Convocatorias Oportunidades de Inversión. Retrieved from http://www.upme.gov.co/Memorias%20Convocatoria%20Redes%20de%20Alto%20Voltaje/UPME_Alberto_Rodriguez.pdf
- Schneider. (2012a). MiCOM H16x. Retrieved from https://download.schneider-electric.com/files?p_enDocType=User+guide&p_File_Id=25609922&p_File_Name=H16x_EN_GL_C24.pdf&p_Reference=H16x_EN_GL_C24
- Schneider. (2012b). MiCOM H36x. Retrieved from https://download.schneider-electric.com/files?p_enDocType=User+guide&p_File_Id=25609964&p_File_Name=H36x_EN_GL_C24.pdf&p_Reference=H36x_EN_GL_C24

- Schneider. (2014). MiCOM H18x. Retrieved from https://download.schneider-electric.com/files?p_enDocType=User+guide&p_File_Id=1187482117&p_File_Name=H18x_EN_GL_A11.pdf&p_Reference=H18x_EN_GL_A11
- Sidhu, T., Kanabar, M., Injeti, S., & Parikh, P. (2010). Packet Scheduling of GOOSE Messages in IEC 61850 based Substation Intelligent Electronic Devices (IEDs). *IEEE PES General Meeting*.
- Siemens. (2011). High Voltage Bay Control Unit 6MD66x. Retrieved from https://www.downloads.siemens.com/download-center/Download.aspx?pos=download&fct=getasset&id1=DLA06_762
- Siemens. (2013a). Input / Output Unit with Local Control 6MD63. Retrieved from https://www.downloads.siemens.com/download-center/Download.aspx?pos=download&fct=getasset&mandator=ic_sg&id1=DLA06_1024
- Siemens. (2013b). Protección de distancia, diferencial de línea y Gestión del interruptor para disparo mono/tripolar 7SA87, 7SD87, 7SL87, 7VK87. Retrieved from <https://support.industry.siemens.com/cs/document/109740238/siprotec-5-7sa87-7sd87-7sl87-7vk87-digsi-device-drivers-v7-3x?dti=0&lc=en-WW>
- Siemens. (2013c). Distance Protection, Line Differential Protection, and Overcurrent Protection for 3-Pole Tripping 7SA82, 7SD82, 7SL82, 7SA84, 7SD84, 7SA86, 7SD86, 7SL86, 7SJ86. Retrieved from https://www.downloads.siemens.com/download-center/download?DLA07_3074
- Siemens. (2013d). RUGGEDCOM M2100. Retrieved from https://www.plcsystems.ru/catalog/ruggedcom/doc/m2100_installationguide.pdf
- Siemens. (2013e). RUGGEDCOM RS940G. Retrieved from https://www.plcsystems.ru/catalog/ruggedcom/doc/rs940g_installationguide.pdf

Siemens. (2014). *Merging Units* 7SC805 Manual. Retrieved from

https://www.google.com.co/url?sa=t&rct=j&q=&esrc=s&source=web&cd=4&cad=rja&uact=8&ved=0ahUKEwjem_GbirzWAhVGQyYKHeQbDhEQFgg4MAM&url=https%3A%2F%2Fwww.downloads.siemens.com%2Fdownload-center%2FDownload.aspx%3Fpos%3Ddownload%26fct%3Dgetasset%26mandator%3Dic_sg%26id1%3DDLA07_2781&usg=AFQjCNHqkDM4mSgM7JVy4qUh_F6Ez2fziw

Siemens. (2015a). RUGGEDCOM RSG2100. Retrieved from

https://cache.industry.siemens.com/dl/files/219/82166219/att_835190/v1/RSG2100_Hardware-Installation-Guide_EN.pdf

Siemens. (2015b). *Merging Units* 6MU805 Manual. Retrieved from

https://www.downloads.siemens.com/download-center/download?DLA09_668

Siemens. (2016a). SIPROTEC 5 High-Voltage Bay Controller 6MD85/86. Retrieved from

https://www.downloads.siemens.com/download-center/Download.aspx?pos=download&fct=getasset&mandator=ic_sg&id1=DLA07_3561

Siemens. (2016b). Overcurrent Protection 7SJ82/7SJ85. Retrieved from

https://www.google.com.co/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwibylXuWJ_XAhUJ5SYKHUa8BW0QFggIMAA&url=https%3A%2F%2Fwww.downloads.siemens.com%2Fdownload-center%2Fdownload%3FDLA07_3610&usg=AOvVaw3K7vWZws8k9a1z4bfw75-z

Siemens. (2016c). Low-Impedance Busbar Protection 7SS85. Retrieved from

<http://w3.siemens.com/smartgrid/global/en/products-systems-solutions/protection/busbar-protection/pages/7ss85.aspx>

Siemens. (2016d). Transformer Differential Protection 7UT82, 7UT85, 7UT86, 7UT87.

Retrieved from

https://www.google.com.co/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwimlsf3wZ_XAhWLSCYKHSFOBtkQFggIMAA&url=https%3A%2F%2Fwww.downloads.siemens.com%2Fdownload-center%2FDownload.aspx%3Fpos%3Ddownload%26fct%3Dgetasset%26id1%3D DLA07_3070&usg=AOvVaw3mhc0Alun5IA4bR9cPRXIL

Skendzic, V., Ender, I., & Zweigle, G. (2007). IEC 61850-9-2 process bus and its impact on power system protection and control reliability. *Schweiter Engineering Laboratories*.

Sugeng, W., Istiyanto, J., Mustofa, K., & Ashari, A. (2015). The Impact of QoS Changes towards Network Performance. *International Journal of Computer Networks and Communications Security*, 3(2).

Tan, J. C., & Luan, W. (2011). IEC 61850 BASED SUBSTATION AUTOMATION SYSTEM ARCHITECTURE DESIGN.

Tournier, J. C., & Werner, T. (2010). A Quantitative Evaluation of IEC61850 Process Bus Architectures.

United States Department of Agriculture. (2001). Design Guide for Rural Substations.

Retrieved from https://www.rd.usda.gov/files/UEP_Bulletin_1724E-300.pdf

UPME. (2009). SELECCIÓN DE UN INVERSIONISTA Y UN INTERVENTOR PARA EL DISEÑO, ADQUISICIÓN DE LOS SUMINISTROS, CONSTRUCCIÓN, OPERACIÓN Y MANTENIMIENTO DE LA SUBESTACIÓN NUEVA ESPERANZA 500/230 kV (TRANSFORMADOR 450 MVA – 500/230 kV) Y LAS LÍNEAS DE TRANSMISIÓN ASOCIADAS. Retrieved from <http://www1.upme.gov.co/PromocionSector/ConvocatoriasSistemaTransmisionNa>

cional/UPME%2001-2008/Anexo%201%20-

%20Descripcion_Proyecto_Nueva_Esperanza.pdf

UPME. (2010). Esquema unifilar subestación Bolívar 220kV. Retrieved from

http://www1.upme.gov.co/PromocionSector/InformacionInversionistas/Documents/UPME%20STR%2010-2015/Figura_3_Adenda_2.pdf

UPME. (2012). SELECCIÓN DE UN INVERSIONISTA Y UN INTERVENTOR PARA EL DISEÑO, ADQUISICIÓN DE LOS SUMINISTROS, CONSTRUCCIÓN, OPERACIÓN Y MANTENIMIENTO DE LA SUBESTACIÓN TERMOCOL 220 kV Y LAS LÍNEAS DE TRANSMISIÓN ASOCIADAS. Retrieved from

http://www.upme.gov.co/Convoca2010/022010/Oficial/Anexo_1_Descripcion_del_Proyecto_Termocol.pdf

UPME. (2013). Esquema unifilar subestación Chinú 500kV. Retrieved from

http://www.upme.gov.co/Convoca2013/072013/Documentos/Figura_3_Unifilar_Chinu_500kV.pdf

UPME. (2015). Plan de Expansión de Referencia Generación – Transmisión 2012-2025.

Retrieved from

http://www.upme.gov.co/docs/plan_expansion/2012/plan_completo_2012-2025.pdf

UPME. (2016). PLAN DE EXPANSIÓN DE REFERENCIA GENERACIÓN –

TRANSMISIÓN 2016 – 2030. Retrieved from

https://www.google.com.co/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwiivoyQ3qXXAhXHOyYKHecGDlgQFggnMAA&url=http%3A%2F%2Fwww.upme.gov.co%2FDocs%2FPlan_Expansion%2F2016%2FPlan_GT_2016_2030%2FPlan_GT_2016_2030_Final_V1_12-12-2016.pdf&usg=AOvVaw220c-1mMSW7XgYW74lvGMo

- Van Rensburg, M., Dolezilek, D., & Dearien, J. (2015). Case Study: Using IEC 61850 Network Engineering guideline Test Procedures to Diagnose and Analyze Ethernet Network Installations. *PAC World Africa Conference*.
- Varga, A. (2003). OMNET++ Academic Public License. Retrieved from <https://omnetpp.org/intro/license>
- Varga, A., & OpenSim Ltd. (2016). OMNeT++ Simulation Manual. Version 5.2. Retrieved from <https://omnetpp.org/doc/omnetpp/SimulationManual.pdf>
- Vico, J., Smith, T., & Hunt, R. (2010). Fully Utilizing the Intelligent Electronic Device Capability to Reduce Wiring in Industrial Electric Distribution Substations. *2010 IEEE Industry Applications Society Annual Meeting*.
- Vidiati, M., Asadi, M., Shahb azi, B., Farzalizadeh, S., Shariati, M., & Rassaie, M. (2008). A New Approach for Determination of the Communication Buses Architecture Based on IEC 6180 in Substation Automation System. *International Symposium on Power Electronics, Electrical Drives, Automation and Motion, 2008. SPEEDAM 2008*.
- Virdis, A., Stea, G., & Nardini, G. (2014). SimuLTE - A modular system-level simulator for LTE/LTE-A networks based on OMNeT++. *2014 4th International Conference On Simulation And Modeling Methodologies, Technologies And Applications (SIMULTECH)*.
- Vizimax. (1997a). TECHNICAL NOTICE SMRB2. Retrieved from <http://ecti.co.th/wp-content/uploads/2017/06/SMRB2-NTRB-0326-Rev-0.pdf>
- Vizimax. (1997b). TECHNICAL NOTICE SMTS. Retrieved from <http://ecti.co.th/wp-content/uploads/2017/06/SMTS-NTTS-0323-Rev-0.pdf>
- Vizimax. (1997c). TECHNICAL NOTICE SMTU. Retrieved from <http://ecti.co.th/wp-content/uploads/2017/06/SMTU-NTTU-0324-Rev-0.pdf>

- Vizimax. (1998). TECHNICAL NOTICE SMRS. Retrieved from <http://ecti.co.th/wp-content/uploads/2017/06/SMRS2-NTRE-0322-Rev-0.pdf>
- Vizimax. (2012a). RightWON Engine Standalone. Retrieved from <http://ecti.co.th/wp-content/uploads/2017/06/RWE044011-SP-en-RightWONEngineStandaloneV2.1.pdf>
- Vizimax. (2012b). RightWON Engine Rackmount, IEC 61850-3 certified. Retrieved from <https://www.vizimax.com/support/download?id=209>
- Vizimax. (2017). STAND ALONE *MERGING UNITS* (AMU) DATASHEET. Retrieved from <http://ecti.co.th/wp-content/uploads/2017/06/MGU010000-SP-en-20170505-Stand-Alone-Merging-Unit-Datasheet.pdf>
- Yang, S., Yang, H., Ahn, Y., & Kim, Y. (2012). Performance Analysis of IEC 61850 based Substation. *14th International Conference on Advanced Communication Technology (ICACT), 2012.*

Anexo A: Correspondencia intercambiada con fabricantes durante la búsqueda de información primaria

Vizimax

17/11/2017

Gmail - Respuesta Vizimax



Francisco Javier Serna Alzate <fjsema89@gmail.com>

Respuesta Vizimax

4 mensajes

Andrea Gutierrez <agutierrez@vizimax.com>
Para: fjsema89@gmail.com

22 de agosto de 2017, 11:04

Estimado Francisco,

Muchas gracias por tu interés en nuestra tecnología.

El MTBF/MTTF de las unidades PMU/AMU de VIZIMAX es 12,56 años, 10,14 años para el RightWON Plus, 23,28 años para el RightWON Engine y 26,47 años para el RightWON Satellite. Estos calculados según el US Military Handbook MIL-HDBK-217F notice2 level (GB Ground Benign) + (3%/KPoHr Non-er Military).

La vida útil de una unidad es aproximadamente tres veces su MTBF.

El MTTR es menos de 1 hora siempre que una unidad de repuesto esté disponible en el sitio.

As for the processing times, it all depends what type of processing you have in mind!!!!

A RightWON-based modular architecture is a way of guaranteeing the processing won't choke the way the processing of very large centralized RTUs sometimes does, since the number of CPUs on duty will be proportional to the number of I/Os. This being said, if the question is whether a RightWON will be fast enough to implement a protection function within an IEC-61850/GOOSE architecture, the answer is no. The RightWON remains an RTU/PLC. Ideal for specialized SCADA/RTU applications, PLC apps requiring a web interface and protocol gateways, but not to implement the ultra-high speed GOOSE-based protections you would never implement in an RTU.

En nuestra página web puedes descargar las fichas técnicas.

Feliz tarde,

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view-pt&q=vizimax&q=TRUE&search=query&th=15ebab5676da586c&siml=15e0aaf0b0b5ded3&siml=15eaf78961eae2a...> 1/5

17/11/2017

Gmail - Respuesta Vizimax

Andrea Gutierrez, M.Eng.

Regional Manager/Gerente Regional



2284, rue de la Province

Longueuil (Québec) J4G 1G1 CANADA

C: +57 3155959202 | agutierrez@vizimax.com



www.vizimax.com

AVIS DE CONFIDENTIALITÉ

Ce courriel est confidentiel, peut être protégé par le secret professionnel et est à l'usage exclusif du destinataire. Toute autre personne est, par les présentes, avisée qu'il lui est strictement interdit de le diffuser, de le distribuer ou de le reproduire. Si ce message ne vous est pas destiné, veuillez en informer immédiatement l'expéditeur par courriel et en détruire toute copie. Merci.

CONFIDENTIALITY NOTICE

This e-mail message is confidential, may be privileged and is intended for the exclusive use of the addressee. Any other person is strictly prohibited from disclosing, distributing or reproducing it. If you are not the intended recipient, please inform the sender by return e-mail immediately and delete any copies of this message. Thank you.



SVP pensez à l'environnement avant d'imprimer ce message / Please consider the environment before printing this message

Francisco Javier Serna Alzate <fjserna89@gmail.com>
Para: Andrea Gutierrez <agutierrez@vizimax.com>

23 de septiembre de 2017, 11:01

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=vizimax&qv=true&search=query&th=15ebab5678da586c&siml=15e0aaffb0b5ded3&siml=15eaf78961eae2a...> 2/5

17/11/2017

Gmail - Respuesta Vizimax

Cordial saludo, Andrea

Muchas gracias por su respuesta. Si es posible, quisiera aclarar un poco sobre la pregunta del tiempo de procesamiento: Me interesa conocer cuánto tiempo transcurre entre el instante en que se recibe (o muestrea) un valor análogo de tensión o corriente y el instante en que se publica el mensaje Sampled Value.

Me interesa conocer este tiempo para cada referencia de Merging Unit Vizimax

Agradezco mucho su amable y pronta respuesta.
[El texto citado está oculto]

—
FRANCISCO JAVIER SERNA ALZATE

Profesional Junior I+D+i

Tel: (574) 444 12 11 ext. 157

CORPORACIÓN CENTRO DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO DEL SECTOR ELÉCTRICO

Camera 46 56-11 Edificio TecnoParque Pisos 13 y 16 Medellín

Fax: (574) 444 04 60

Andrea Gutierrez <agutierrez@vizimax.com>
Para: Francisco Javier Serna Alzate <fjserna89@gmail.com>

25 de septiembre de 2017, 15:23

Estimado Francisco,

Es con mucho gusto!

A 256 muestras/ciclo el retardo promedio es de aproximadamente 366 μ s. medido con la ayuda de un Omicron.

Saludos,

17/11/2017

Gmail - Respuesta Vizimax

Andrea Gutierrez, M.Eng.

Regional Manager/Gerente Regional



2284, rue de la Province

Longueuil (Québec) J4G 1G1 CANADA

C: +57 3155959202 | agutierrez@vizimax.comwww.vizimax.com

AVIS DE CONFIDENTIALITÉ

Ce courriel est confidentiel, peut être protégé par le secret professionnel et est à l'usage exclusif du destinataire. Toute autre personne est, par les présentes, avisée qu'il lui est strictement interdit de le diffuser, de le distribuer ou de le reproduire. Si ce message ne vous est pas destiné, veuillez en informer immédiatement l'expéditeur par courriel et en détruire toute copie. Merci.

CONFIDENTIALITY NOTICE

This e-mail message is confidential, may be privileged and is intended for the exclusive use of the addressee. Any other person is strictly prohibited from disclosing, distributing or reproducing it. If you are not the intended recipient, please inform the sender by return e-mail immediately and delete any copies of this message. Thank you.



SVP pensez à l'environnement avant d'imprimer ce message / Please consider the environment before printing this message

From: Francisco Javier Serna Alzate [<mailto:fjserna89@gmail.com>]**Sent:** September 23, 2017 11:01 AM

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=vizimax&q=1&search=query&th=15e0ab5678da586c&siml=15e0aafb0b5ded3&siml=15eaf78961eae2a...> 4/5

17/11/2017

Gmail - Respuesta Vizimax

To: Andrea Gutierrez
Subject: Re: Respuesta Vizimax

[El texto citado está oculto]

Francisco Javier Serna Alzate <fjserna89@gmail.com>
Para: Andrea Gutierrez <agutierrez@vizimax.com>

25 de septiembre de 2017, 15:23

Muchas gracias Andrea por su pronta respuesta

[El texto citado está oculto]

Siemens

17/11/2017

Gmail - Tiempos de respuesta de algunos dispositivos de la familia Ruggedcom



Francisco Javier Serna Alzate <fjserna89@gmail.com>

Tiempos de respuesta de algunos dispositivos de la familia Ruggedcom

2 mensajes

Paulina Henao <dphenaov@elicoogrp.com>
 Para: fjserna89@gmail.com
 Cc: Jorge Toro <jmtoro@elicoogrp.com>

23 de agosto de 2017, 15:14

Cordial Saludo Señor Francisco,

Con base en la solicitud que nos hace a través de la plataforma de consulta Solution Partner Siemens, Queremos agradecerle principalmente por su interés en las tecnologías de la Marca Siemens y por permitimos apoyarlo en su solicitud. de acuerdo a su pregunta tenemos la siguiente información:

En la familia Ruggedcom el tiempo de respuesta está entre 5 y 7 micro segundos, en cuanto al envío de paquetes este se mide en la salida del paquete y el regreso del mismo juntos, Es decir, se mide el tiempo desde que sale el paquete hasta que llega la respuesta

Saludos,
 Paulina Henao Valdés
 Ingeniera Comercial
 3146817942
dphenaov@elicoogrp.com
 (57)+4 4447441
 Cra 48#100 sur-72
 La Estrella Antioquia

Francisco Javier Serna Alzate <fjserna89@gmail.com>
 Para: Paulina Henao <dphenaov@elicoogrp.com>

23 de septiembre de 2017, 11:34

Cordial saludo, Paulina

Muchas gracias por su amable respuesta. Si es posible, quisiera hacerle una consulta adicional: este tiempo que usted menciona de entre 5 y 7 microsegundos es el mismo tiempo que tarda la Merging Unit en formar el mensaje Sampled Value? en otras palabras, me interesa conocer el tiempo que transcurre entre el instante en que la Merging Unit recibe (o muestrea) un valor análogo de tensión y corriente y el instante en que es enviado el mensaje Sampled Value.

Adicionalmente, le agradecería si pudiera indicarme la tasa de errores de la Merging Unit, puede ser el Bit Error Rate, o cualquier otra métrica que indique qué tan probable es que la Merging Unit envíe un mensaje con datos incorrectos o no sincronizados.

De nuevo muchísimas gracias por su colaboración. Estaré atento a su pronta respuesta.
 [El texto citado está oculto]

—
 FRANCISCO JAVIER SERNA ALZATE

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=Siemens&qs=true&search=query&th=15eaf96cea89ffa2&siml=15e10bbbfa66af8&siml=15eaf96cea89ffa2> 1/2

17/11/2017

Gmail - Tiempos de respuesta de algunos dispositivos de la familia Ruggedcom

Profesional Junior I+D+i

Tel: (574) 444 12 11 ext. 157

CORPORACIÓN CENTRO DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO DEL SECTOR ELÉCTRICO

Carrera 40 56-11 Edificio TecnoParque Pisos 13 y 16 Medellín

Fax: (574) 444 04 60

17/11/2017

Gmail - SIEMENS IED reliability indexes



Francisco Javier Serna Alzate <fjserna89@gmail.com>

SIEMENS IED reliability indexes

2 mensajes

Francisco Javier Serna Alzate <fjserna89@gmail.com>
Para: support.energy@siemens.com

27 de diciembre de 2016, 4:23

Dear Siemens support team:

I would like to know some information regarding to the reliability of the following IED devices:

- SIPROTEC 8MD85
- SIPROTEC 8MD86
- SIPROTEC 7SL87
- SIPROTEC 7SJ82
- SIPROTEC 7SA86
- SIPROTEC 7UT82.

Particularly, I would like to know indexes such as the Mean Time To Failure - MTTF or Mean Time Before Failure - MTBF, Mean Time To Repair - MTTR and the expected lifetime for each device.

Thank you very much for your reponse.

--

FRANCISCO JAVIER SERNA ALZATE

Francisco Javier Serna Alzate <fjserna89@gmail.com>
Para: support.energy@siemens.com

27 de diciembre de 2016, 5:10

I am also interested in the indexes MTTF, MTBF, MTTR and lifetime for the following Merging Units:

- SIPROTEC 8MU80
- SIPROTEC 7SC805

It would be very helpful if you could please provide this information.

Thank you very much for your kind response.

[El texto citado está oculto]

FRANCISCO JAVIER SERNA ALZATE

Alstom

17/11/2017

Gmail - Informaciones MU320



Francisco Javier Serna Alzate <fjserna89@gmail.com>

Informaciones MU320

5 mensajes

SILVA Danilo <danilo.silva@alstom.com>
Para: "fjserna89@gmail.com" <fjserna89@gmail.com>

11 de enero de 2017, 8:37

Estimado Francisco buen día,

reciba mis cordiales saludos desde Brasil y disculpas por la demora de una respuesta.

Según has solicitado a través del nuestro Contact Center, te envío los datos de MTBF del equipo MU320:

MTBF = 132 años.

Cualquier otra necesidad estamos a disposición.

Saludos cordiales,

Danilo Silva

Support Analyst - Reason Product Line
GE Energy Connections
Grid Solutions

T +55 48 2108-0300 | F +55 48 2108-0310
LinkedIn | Twitter
www.gegridsolutions.com

855 Delminda Silveira | Florianopolis, SC 88025-500 Brazil

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=contact.centre%40ge.com&qe=true&search=query&th=15f35033f8ef95a3&siml=1598d0fbad5edb4b&siml=1...> 1/5

17/11/2017

Gmail - Informaciones MU320

CONFIDENTIALITY : This e-mail and any attachments are confidential and may be privileged. If you are not a named recipient, please notify the sender immediately and do not disclose the contents to another person, use it for any purpose or store or copy the information in any medium.

Francisco Javier Serna Alzate <fjserna89@gmail.com>
Para: SILVA Danilo <danilo.silva@alstom.com>

23 de septiembre de 2017, 11:18

Cordial saludo, Danilo

Muchas gracias por su respuesta. Si es posible, quiero consultarle alguna información adicional:

Con respecto a las Merging Units MU-Agile XMU800, MU-Agile XMU820, Reason MU 320 y MU Agile AMU, me interesa conocer la siguiente información:

- tiempo de procesamiento: tiempo que pasa entre el instante en que se recibe (o muestrea) un valor análogo de tensión o corriente y el instante en que se envía el mensaje Sampled Value
- Velocidad de transmisión: tasa máxima de datos en Mbps que puede enviar cada Merging Unit
- Bit Error Rate: o cualquier otra medida de la probabilidad de enviar un Sampled Value con información errónea o sin sincronización
- Las Merging Units cuentan con puerto para sincronización mediante IRIG-B?

Agradezco mucho su colaboración y pronta respuesta.

[El texto citado está oculto]

FRANCISCO JAVIER SERNA ALZATE

Profesional Junior I+D+i

Tel: (574) 444 12 11 ext. 157

CORPORACIÓN CENTRO DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO DEL SECTOR ELÉCTRICO

Camera 46 56-11 Edificio TecnoParque Pisos 13 y 16 Medellín

Fax: (574) 444 04 60

Francisco Javier Serna Alzate <fjserna89@gmail.com>
Para: contact.centre@ge.com, danilo.silva@ge.com

23 de septiembre de 2017, 11:39

----- Mensaje reenviado -----

De: Francisco Javier Serna Alzate <fjserna89@gmail.com>

Fecha: 23 de septiembre de 2017, 11:18

Asunto: Re: Informaciones MU320

Para: SILVA Danilo <danilo.silva@alstom.com>

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=contact.centre%40ge.com&qe=true&search=query&th=15f35033f6ef95a3&siml=1596dbf6ad6edb4b&siml=1...> 2/5

17/11/2017

Gmail - Informaciones MU320

[El texto citado está oculto]
 [El texto citado está oculto]

Francisco Javier Serna Alzate <fjserna89@gmail.com>
 Para: SILVA Danilo <danilo.silva@alstom.com>

19 de octubre de 2017, 9:21

Cordial saludo Danilo

Agradezco sinceramente su amable respuesta. Espero no molestar si le consulto por unos parámetros adicionales:

Para las Merging Units de referencia COSI-MU, MU-320DS, MU 320-TM, AGILE XMU 800, AGILE XMU 820 y REASON MU 320, necesito conocer información sobre los siguientes parámetros:

- **Tiempo de procesamiento:** tiempo que transcurre entre el instante en que se recibe (o muestrea) un valor análogo de tensión o corriente y el instante en que se envía el mensaje Sampled Value
- **Velocidad de transmisión:** tasa de envío de datos en Mbps máxima que puede enviar cada Merging Unit
- **Bit Error Rate:** tasa de bits incorrectos que envía cada Merging Unit. También me es útil conocer cualquier otra medida de la probabilidad de que la merging Unit envíe un dato incorrecto o no sincronizado
- Las Merging Units cuentan con un puerto dedicado para sincronización mediante IRIG-B?

Adicionalmente, para el controlador de bahía de referencia MiCOM264, requiero los siguientes parámetros:

- **Tiempo de procesamiento:** tiempo que transcurre entre el instante en que se recibe un mensaje Sampled Value, se decodifica, se toma una decisión y el instante en que se envía un mensaje GOOSE con una orden de control. También podría ser el tiempo que el dispositivo tarda en elaborar una trama GOOSE y enviarla a la red.
- **Velocidad de transmisión:** tasa de envío de datos en Mbps máxima que puede enviar cada controlador de bahía
- **Bit Error Rate:** tasa de bits incorrectos o probabilidad de error en bits de los mensajes enviados por el controlador de bahía.

Si es posible estimar los mismos parámetros para el Switch MiCOM H-Series, me sería de gran utilidad.

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=contact.centre%40ge.com&q=15f35033f8ef95a3&siml=1598dbfbad5edb4b&siml=1...> 3/5

17/11/2017

Gmail - Informaciones MU320

Muchas gracias por su pronta y amable respuesta.

2017-01-11 8:37 GMT-05:00 SILVA Danilo <danilo.silva@alstom.com>:
[El texto citado está oculto]

--

FRANCISCO JAVIER SERNA ALZATE

Profesional Junior I+D+i

Tel: (574) 444 12 11 ext. 157

CORPORACIÓN CENTRO DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO DEL SECTOR ELÉCTRICO

Carrera 46 56-11 Edificio TecnoParque Pisos 13 y 16 Medellín

Fax: (574) 444 04 60

Mail Delivery Subsystem <mailer-daemon@googlemail.com>
Para: fjserna89@gmail.com

19 de octubre de 2017, 9:22



El mensaje no se ha podido enviar

El mensaje no se ha podido enviar a danilo.silva@alstom.com porque el servidor remoto está mal configurado. Consulta los detalles técnicos que se indican a continuación para obtener más información.

Esta es la respuesta del servidor remoto:

550 5.4.1 [danilo.silva@alstom.com]: Recipient address rejected: Access denied [VE1EUR03FT040.eop-EUR03.prod.protection.outlook.com]

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=contact.centre%40ge.com&q=true&search=query&th=15f35033f8ef95a3&siml=1598dbf6ad6edb4b&siml=1...> 4/5

17/11/2017

Gmail - Informaciones MU320

Final-Recipient: rfc822: danilo.silva@alstom.com
Action: failed
Status: 5.4.1
Remote-MTA: dns: alstom-com.mail.protection.outlook.com. (213.199.154.138, the server for the domain alstom.com)
Diagnostic-Code: smtp: 550 5.4.1 [danilo.silva@alstom.com]: Recipient address rejected: Access denied [VE1EUR03FT040.eop-EUR03.prod.protection.outlook.com]
Last-Attempt-Date: Thu, 19 Oct 2017 07:22:05 -0700 (PDT)

----- Mensaje reenviado -----
From: Francisco Javier Serna Alzate <fjserna89@gmail.com>
To: SILVA Danilo <danilo.silva@alstom.com>
Cc:
Bcc:
Date: Thu, 19 Oct 2017 09:21:53 -0500
Subject: Re: Informaciones MU320
Cordial saludo Danilo

Agradezco sinceramente su amable respuesta. Espero no molestar si le consulto por unos parámetros adicionales:

Para las Merging Units de referencia COSI-MU, MU-320DS, MU 320-TM, AGILE XMU 800, ----- Message truncated -----

ABB

17/11/2017

Gmail - FW: Requisición información MU



Francisco Javier Serna Alzate <fjserna89@gmail.com>

FW: Requisición información MU

4 mensajes

Diana Becerra <diana.becerra@co.abb.com>
Para: "fjserna89@gmail.com" <fjserna89@gmail.com>

25 de septiembre de 2017, 15:47

Buenas tardes Francisco

Me permito presentarme, soy la representante de los equipos para subestaciones digitales, me gustaría consultar su disponibilidad de tiempo y número telefónico para conversar y conocer la solución en la que está interesado.

Estaré atenta a su respuesta,

Cordial saludo,

ABB

—
Diana Liseth Becerra Cabieles

Sales Specialist

Power Grid Division

Grid Automation

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38gD1M8g.es.&view=pt&q=abb&qe=true&search=query&th=15fa806ca2e0caeb&siml=15ebacbe0e24e54d&siml=15f34d373cdd1459&...> 1/12

17/11/2017

Gmail - FW: Requisición información MU

ABB Ltda

Cra 100 No. 25D-81

Sede Industrial

Bogotá D.C, Colombia

Phone: +(57 1) 4178000 Ext. 2347

Mobile: +(57) 3134520882

email: diana.becerra@co.abb.com

-



From: Jhon Alexander. Yaya**Sent:** Monday, September 25, 2017 8:56 AM**To:** Jonathan Lopez <jonathan.lopez@co.abb.com>; Diana Becerra <diana.becerra@co.abb.com>**Subject:** FW: Requisición información REF 615

Estimados Jonathan / Diana,

Es posible validar si los equipos mostrados abajo corresponden a PGGA.

SAM600-CT, SAM600-VT y CP-MU.

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&q=1true&search=query&th=15fa806ca2e0caeb&siml=15ebacbe0e24e54d&siml=15f34d373odd1459&...> 2/12

17/11/2017

Gmail - FW: Requisición Información MU

Atentamente,**Jhon Yaya**

Jhon Alexander Yaya

Coordinador de Ofertas – Media Tensión

ABB Ltda - EPMV

Av. Carrera 45 # 108 – 27, Torre 1, Piso 12.

Centro Empresarial Paralelo 108, Bogotá – Colombia.

Teléfono: +57 1 4178000 Ext: 3620

Celular: +57 310 233 2380

jhon.yaya@co.abb.comSolutions for a smarter, safer
and more reliable flow of electricity.[SEE MORE](#)

From: Francisco Nieto

Sent: Monday, September 25, 2017 7:48 AM

To: Jhon Alexander. Yaya <jhon.yaya@co.abb.com><https://mail.google.com/mail/u/0/?ui=2&ik=4fcca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&q=15fa806ca2e0caeb&simi=15ebacbe0e24e54d&simi=15f34d373odd1459&...> 3/12

17/11/2017

Gmail - FW: Requisición Información MU

Cc: David Paloma <david.paloma@co.abb.com>
Subject: FW: Requisición información REF 615

Estimado Jhon:

Le agradezco me informe quien puede colaborar con la información requerida por esta persona.

Atentamente



Luis Francisco Nieto F.
External Channel Manager
Power Technologies
Avda. Cra 45 No. 108-27, Torre 1, Piso 12
Centro Empresarial Paralelo 108 (Autopista Norte, Bogotá- Colombia)
Tel: +57 (1) 4178000 Ext: 3606, Mobile: (57) 3115617162
email: francisco.nieto@co.abb.com

From: Francisco Javier Serna Alzate [<mailto:fjserna89@gmail.com>]
Sent: Saturday, September 23, 2017 11:27 AM
To: Francisco Nieto <francisco.nieto@co.abb.com>
Subject: Re: Requisición información REF 615

CAUTION: This email originated from outside of the organization. Do not click links or open attachments unless you recognize the sender and know the content is safe.

17/11/2017

Gmail - FW: Requisición Información MU

Cordial saludo, Francisco

Muchas gracias por su amable respuesta. Si es posible, me gustaría aclarar un poco sobre la información que estoy requiriendo:

Para las Merging Units de referencia SAM800-CT, SAM800-VT y CP-MU, necesito conocer información sobre los siguientes parámetros:

- **Tiempo de procesamiento:** tiempo que transcurre entre el instante en que se recibe (o muestrea) un valor análogo de tensión o corriente y el instante en que se envía el mensaje Sampled Value
- **Velocidad de transmisión:** tasa de envío de datos en Mbps máxima que puede enviar cada Merging Unit
- **Bit Error Rate:** tasa de bits incorrectos que envía cada Merging Unit. También me es útil conocer cualquier otra medida de la probabilidad de que la merging Unit envíe un dato incorrecto o no sincronizado
- Las Merging Units cuentan con un puerto dedicado para sincronización mediante IRIG-B?

Agradezco mucho su colaboración y pronta respuesta.

El 28 de agosto de 2017, 9:11, Francisco Nieto <francisco.nieto@co.abb.com> escribió:

Estimados Señores:

De acuerdo con lo solicitado por nuestra WEB, adjunto a la presente encontrará un catálogo de la protección REF 615.

Le agradecemos ver la información y en caso de alguna inquietud, por favor no dude en hacémosla saber.

Atentamente

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&q&true&search=query&th=15fa806ca2e0caeb&simi=15ebacbe0e24e54d&simi=15f34d373odd1459&...> 5/12

17/11/2017

Gmail - FW: Requisición Información MU



Luis Francisco Nieto F.
External Channel Manager
Power Technologies
Avda. Cra 45 No. 108-27, Torre 1, Piso 12
Centro Empresarial Paralelo 108 (Autopista Norte, Bogotá- Colombia)
Tel: +57 (1) 4178000 Ext: 3606, Mobile: (57) 3115617162
email: francisco.nieto@co.abb.com

--

FRANCISCO JAVIER SERNA ALZATE

Profesional Junior I+D+i

Tel: (574) 444 12 11 ext. 157

CORPORACIÓN CENTRO DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO DEL SECTOR ELÉCTRICO

[Carrera 46 56-11 Edificio TecnoParque](#) Pisos 13 y 16 Medellín

Fax: (574) 444 04 80

Francisco Javier Serna Alzate <fjserna80@gmail.com>
Para: Diana Becerra <diana.becerra@co.abb.com>

19 de octubre de 2017, 8:29

Cordial saludo, Diana

Muchas gracias por su respuesta. Mi celular es 312 846 94 12, estoy en Medellín. Me interesa conocer para las Merging Units de referencia SAM600-CT, SAM600-VT y CP-MU, los siguientes parámetros:

- **Tiempo de procesamiento:** tiempo que transcurre entre el instante en que se recibe (o muestrea) un valor análogo de tensión o corriente y el instante en que se envía el mensaje Sampled Value
- **Velocidad de transmisión:** tasa de envío de datos en Mbps máxima que puede enviar cada Merging Unit

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&qs=true&search=query&th=15fa806ca2e0caeb&siml=15ebacbe0e24e54d&siml=15f34d373odd1459&...> 6/12

17/11/2017

Gmail - FW: Requisición Información MU

- **Bit Error Rate:** tasa de bits incorrectos que envía cada Merging Unit. También me es útil conocer cualquier otra medida de la probabilidad de que la merging Unit envíe un dato incorrecto o no sincronizado
- Las Merging Units cuentan con un puerto dedicado para sincronización mediante IRIG-B?

Estos mismos parámetros me interesan para IEDs de Control de las referencias REC 670 y PCM 600.

Agradezco mucho su colaboración y estaré atento su amable respuesta.

[El texto citado está oculto]

Diana Becerra <diana.becerra@co.abb.com>
Para: Francisco Javier Serna Alzate <fjserna89@gmail.com>

9 de noviembre de 2017, 10:45

Hola Tatiana

De acuerdo con nuestra conversación, adjunto te comparto los requerimientos técnicos que nos envió el ingeniero Francisco de la CORPORACIÓN CENTRO DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO DEL SECTOR ELÉCTRICO.

Considero que puedes reunirte con él para validar la información y apoyarte directamente con Gonzalo Ramirez o con la fábrica.

<https://www.youtube.com/watch?v=OM0bHKMZ4Lo>

Saludos,



<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&q=15fa806ca2e0caeb&siml=15ebacbe0e24e54d&siml=15f34d373cdd1459&...> 7/12

17/11/2017

Gmail - FW: Requisición información MU

Diana Liseth Becerra Cabeles

Sales Specialist

Power Grid Division

Grid Automation

ABB Ltda

Cra 100 No. 25D-61

Sede Industrial

Bogotá D.C., Colombia

Phone: +(57 1) 4178000 Ext. 2347

Mobile: +(57) 3134520662

email: diana.becerra@co.abb.com

From: Francisco Javier Serna Alzate [<mailto:fjserna89@gmail.com>]**Sent:** Thursday, October 19, 2017 8:30 AM**To:** Diana Becerra <diana.becerra@co.abb.com>**Subject:** Re: FW: Requisición información MU

CAUTION: This email originated from outside of the organization. Do not click links or open attachments unless you recognize the sender and know the content is safe.

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0ca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&qv=true&search=query&th=15fa806ca2e0caeb&siml=15ebacbe0e24e54d&siml=15f34d373cdd1459&...> 8/12

17/11/2017

Gmail - FW: Requisición Información MU

Cordial saludo, Diana

Muchas gracias por su respuesta. Mi celular es 312 846 94 12, estoy en Medellín. Me interesa conocer para las Mergin Units de referencia SAM800-CT, SAM800-VT y CP-MU, los siguientes parámetros:

- **Tiempo de procesamiento:** tiempo que transcurre entre el instante en que se recibe (o muestrea) un valor análogo de tensión o corriente y el instante en que se envía el mensaje Sampled Value. **ABB:**

- **Velocidad de transmisión:** tasa de envío de datos en Mbps máxima que puede enviar cada Mergin Unit. **ABB:**

- **Bit Error Rate:** tasa de bits incorrectos que envía cada Merging Unit. También me es útil conocer cualquier otra medida de la probabilidad de que la merging Unit envíe un dato incorrecto o no sincronizado. **ABB:**

- Las Mergin Units cuentan con un puerto dedicado para sincronización mediante IRIG-B?

ABB: Las merging units ABB son sincronizables por IRIG-B o PPS (pulsos por segundo), para este tipo de aplicaciones se recomienda usar PPS

Estos mismos parámetros me interesan para IEDs de Control de las referencias REC 670 y PCM 600.

ABB:

Agradezco mucho su colaboración y estaré atento su amable respuesta.

--

FRANCISCO JAVIER SERNA ALZATE

Profesional Junior I+D+i

Tel: (574) 444 12 11 ext. 157

CORPORACIÓN CENTRO DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO DEL SECTOR ELÉCTRICO

[Camera 48 56-11 Edificio TecnoParque Pisos 13 y 16 Medellín](#)

Fax: (574) 444 04 60

Tatiana Arrighi-Crespo <tatiana.arrighi-crespo@co.abb.com>
Para: Francisco Javier Serna Alzate <fjserna89@gmail.com>

10 de noviembre de 2017, 17:22

<https://mail.google.com/mail/u/0/?ui=2&ik=47cfca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&q=1true&search=query&th=15fa806ca2e0cae&siml=15ebacbe0e24e54d&siml=15f34d373cdd1459&...> 9/12

17/11/2017

Gmail - FW: Requisición Información MU

Buenas tardes Francisco J. Serna

Según lo conversado telefónicamente te confirmo que estamos recopilando la información solicitada, con el propósito de atender mejor tu requerimiento si te parece bien podemos programar una pequeña reunión y socializamos la necesidad si te parece bien puede ser este próximo jueves.

Al pie de mi firma encuentras mis datos de contacto. Muchas gracias, quedo atenta a tus comentarios.

Saludos / Regards,

ABB

—

Tatiana Arrighi Crespo

Sales Specialist, Medellín

Power Grid Division

Grid Automation

ABB Ltda.

Carrera 43 A # 8 Sur 15. Oficina 505

Edificio Torre Oviedo

050022 Medellín, Colombia.

Movil: +57 312 4694472

Phone: +57 4 321 5822 Ext. 2215

Fax: +57 4 268 6118

<https://mail.google.com/mail/u/0/?ui=2&ik=4fcdca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&q=trus&search=query&th=15fa806ca2eccaeb&siml=15ebacbe0e24e54d&siml=15f34d373odd1459...> 10/12

17/11/2017

Gmail - FW: Requisición información MU

email: tatiana.arrighi-crespo@co.abb.comabb.com

From: Francisco Javier Serna Alzate [<mailto:fjserna89@gmail.com>]**Sent:** Thursday, October 19, 2017 8:30 AM**To:** Diana Becerra <diana.becerra@co.abb.com>**Subject:** Re: FW: Requisición información MU

CAUTION: This email originated from outside of the organization. Do not click links or open attachments unless you recognize the sender and know the content is safe.

Cordial saludo, Diana

Muchas gracias por su respuesta. Mi celular es 312 846 94 12, estoy en Medellín. Me interesa conocer para las Mergin Units de referencia SAM800-CT, SAM800-VT y CP-MU, los siguientes parámetros:

- **Tiempo de procesamiento:** tiempo que transcurre entre el instante en que se recibe (o muestrea) un valor análogo de tensión o corriente y el instante en que se envía el mensaje Sampled Value. **ABB:**
- **Velocidad de transmisión:** tasa de envío de datos en Mbps máxima que puede enviar cada Mergin Unit. **ABB:**
- **Bit Error Rate:** tasa de bits incorrectos que envía cada Merging Unit. También me es útil conocer cualquier otra medida de la probabilidad de que la merging Unit envíe un dato incorrecto o no sincronizado. **ABB:**

<https://mail.google.com/mail/u/0/?ui=2&ik=4f0fca1926&jsver=C8N38g01M8g.es.&view=pt&q=abb&q=abb&search=query&th=15fa806ca2e0caeb&simi=15ebacbe0e24e54d&simi=15f34d373cdd1459...> 11/12

17/11/2017

Gmail - FW: Requisición Información MU

- Las Merjin Units cuentan con un puerto dedicado para sincronización mediante IRI-G-B?

ABB: Las merjin units ABB son sincronizables por IRI-G-B o PPS (pulsos por segundo), para este tipo de aplicaciones se recomienda usar PPS

Estos mismos parámetros me interesan para IEDs de Control de las referencias REC 670 y PCM 600.

ABB:

Agradezco mucho su colaboración y estaré atento su amable respuesta.

--

FRANCISCO JAVIER SERNA ALZATE

Profesional Junior I+D+i

Tel: (574) 444 12 11 ext. 157

CORPORACIÓN CENTRO DE INVESTIGACIÓN Y DESARROLLO TECNOLÓGICO DEL SECTOR ELÉCTRICO

Carrera 46 56-11 Edificio TecnoParque Pisos 13 y 16 Medellín

Fax: (574) 444 04 60

Anexo B: Diagramas unifilares de las subestaciones analizadas

Subestaciones a 110kV – 230kV

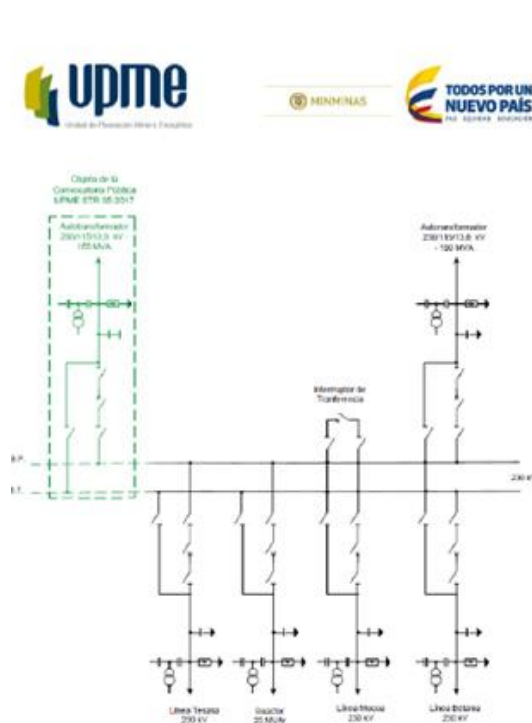


FIGURA 1.
ESQUEMA UNIFILAR SUBESTACIÓN ALTAMIRA 230 kV

Avenida Calle 26 No 69-31 - 91 Torre 1, Oficina 901.
P.O. Box 457 (1) 222 0901 FAX: 221 9537
Línea Gratuita Nacional 0100 911 1729

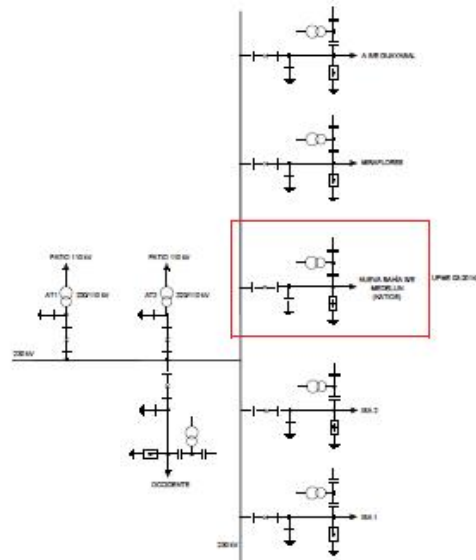
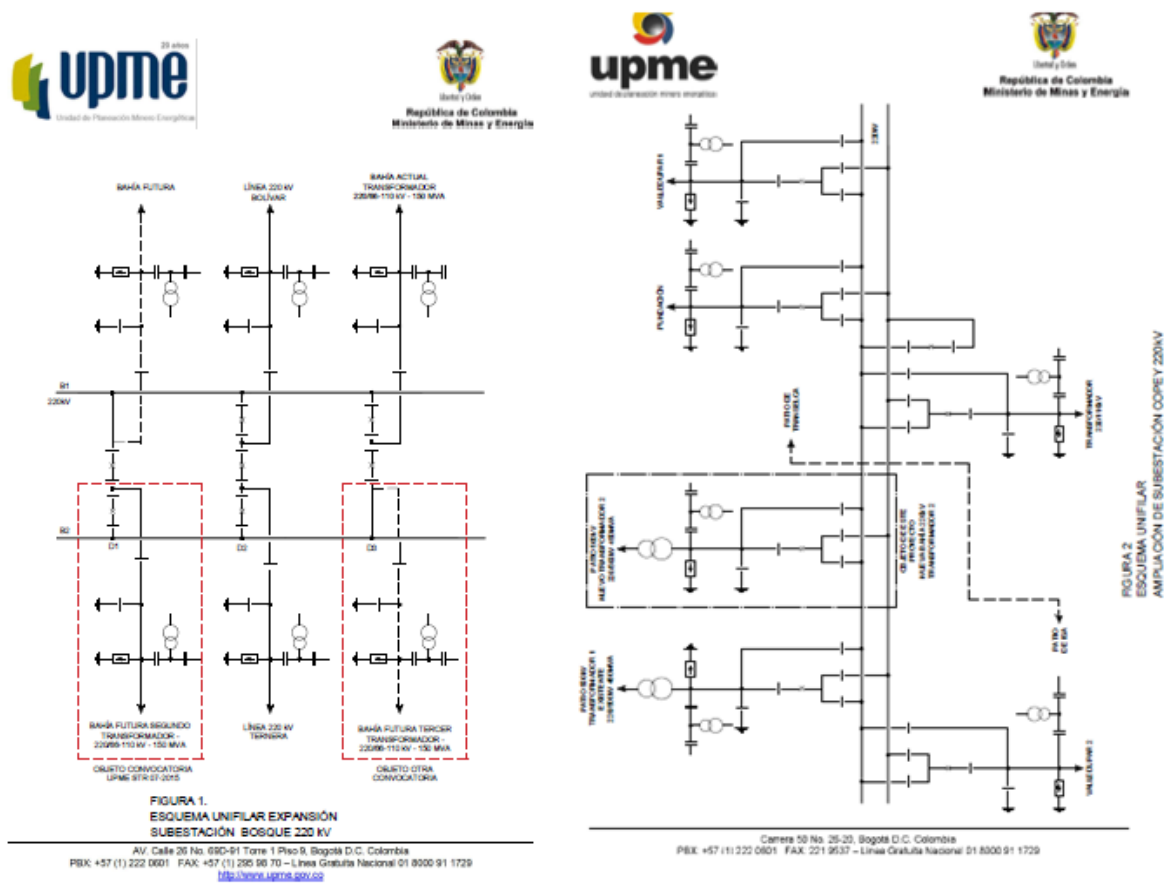


FIGURA 2.
ESQUEMA UNIFILAR
AMPLIACIÓN DE LA SUBESTACIÓN ANCON SUR 230 kV

AV. Calle 26 No. 69-31 Torre 1 Piso 9, Bogotá D.C. Colombia
PBX: +57 (1) 222 0901 FAX: 221 9537 - Línea Gratuita Nacional 01 800 91 1729
<http://www.upme.gov.co>



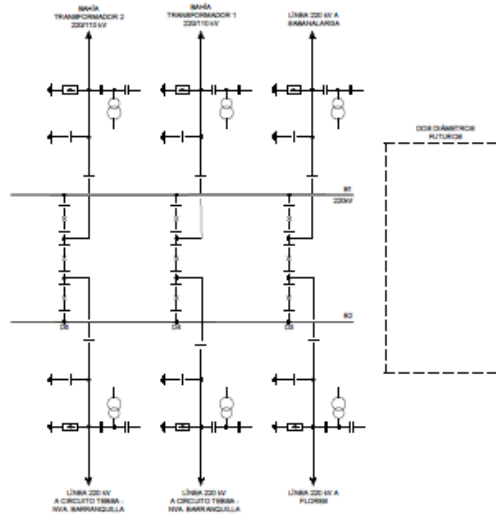


FIGURA 4
ESQUEMA UNIFILAR
SUBESTACIÓN CARACOLI 230 kV

Carrera 50 No. 26-30, Bogotá D.C., Colombia
PBX: +57 (1) 222 0601 FAX: 221 9627 - Línea Gratuita Nacional 01 8000 91 1729
<http://www.upme.gov.co>

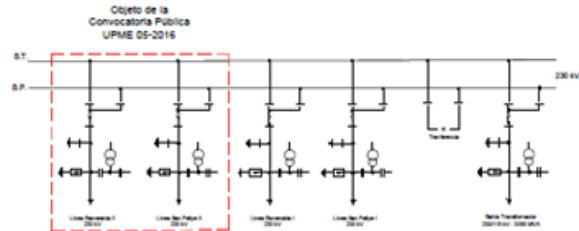


FIGURA 1
ESQUEMA UNIFILAR
SUBESTACIÓN LA ENEA 230 kV

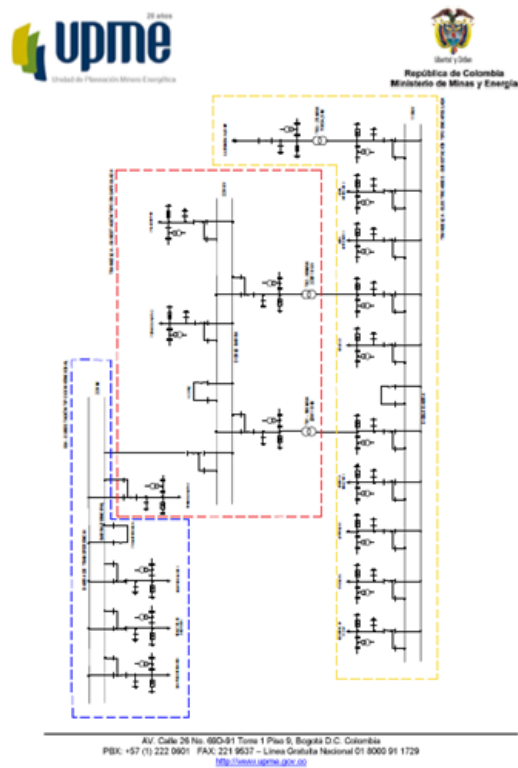


FIGURA 1
ESQUEMA UNIFILAR
SUBESTACIÓN CUESTECITAS 220/110 kV

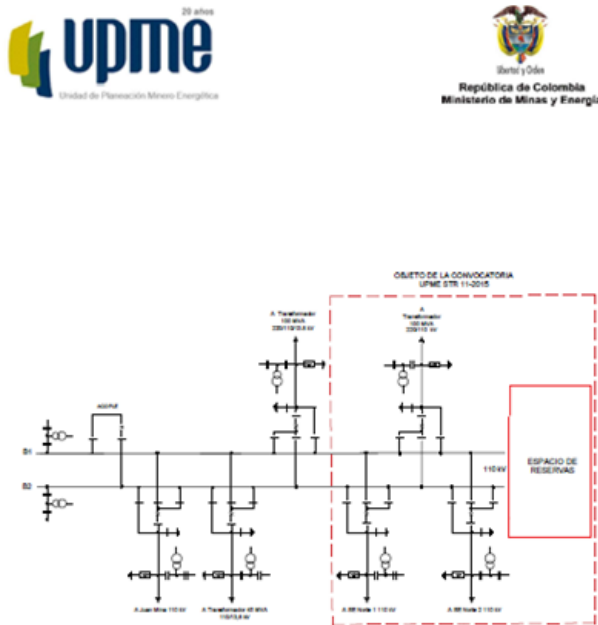


FIGURA 2
ESQUEMA UNIFILAR SUBESTACIÓN
NUEVA BARRANQUILLA 110 kV

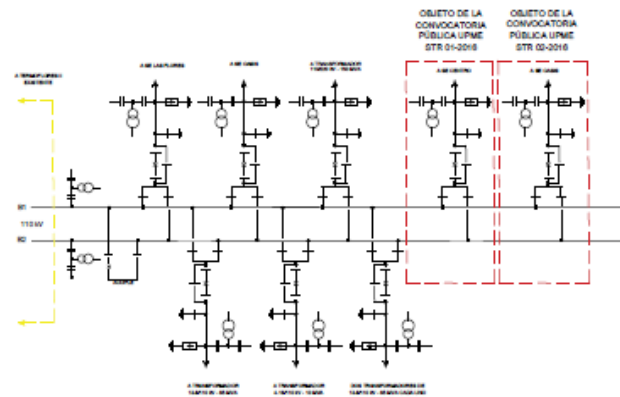


FIGURA 1
ESQUEMA UNIFILAR SUBESTACIÓN
TERMODIORES 110 kV

AV. Calle 26 No. 69D-91 Torre 1 Piso 9, Bogotá D.C. Colombia
PBX: +57 (1) 222 0601 FAX: +57 (1) 205 98 70 – Línea Gratuita Nacional 01 8000 91 1729
<http://www.upme.gov.co>

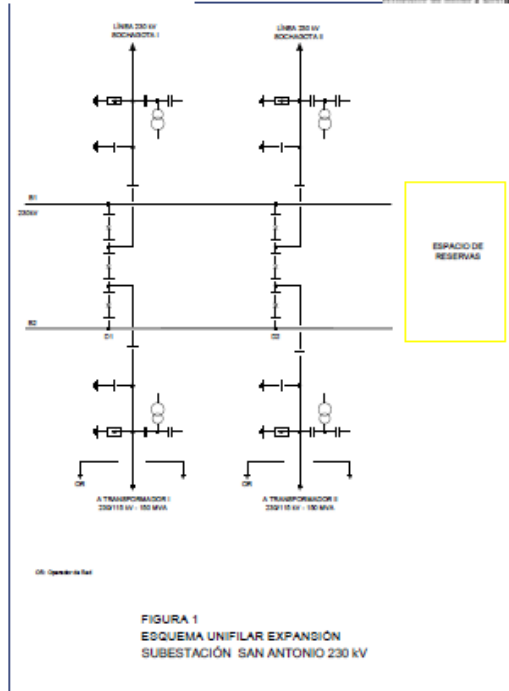


FIGURA 1
ESQUEMA UNIFILAR EXPANSIÓN
SUBESTACIÓN SAN ANTONIO 230 kV

AV. Calle 26 No. 69D-91 Torre 1 Piso 9, Bogotá D.C. Colombia
PBX: +57 (1) 222 0601 FAX: +57 (1) 205 98 70 – Línea Gratuita Nacional 01 8000 91 1729
<http://www.upme.gov.co>

FIGURA 2
ESQUEMA UNIFILAR SUBESTACIÓN
NUEVA BARRANQUILLA 110 kV

Subestaciones a 500kV



República de Colombia
Ministerio de Minas y Energía

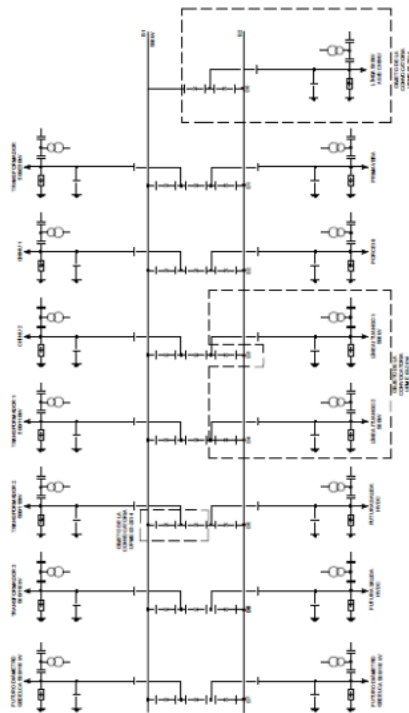


FIGURA 3
ESQUEMA UNIFILAR
SUBESTACIÓN CERROMATOSO 500 kV



República de Colombia
Ministerio de Minas y Energía

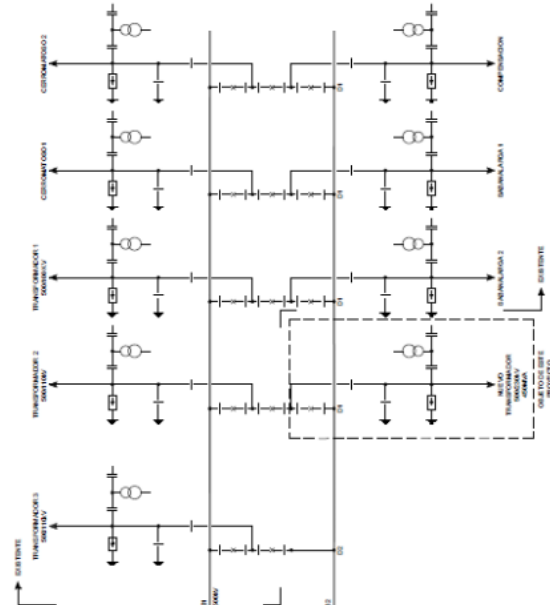


FIGURA 3
ESQUEMA UNIFILAR
SUBESTACIÓN CHINU 500kV

Carrera 50 No. 39-20, Bogotá D.C., Colombia
PBX: +57 (1) 222 0601 FAX: 221 9537 - Línea Gratuita Nacional 01 8000 91 1729
<http://www.upme.gov.co>

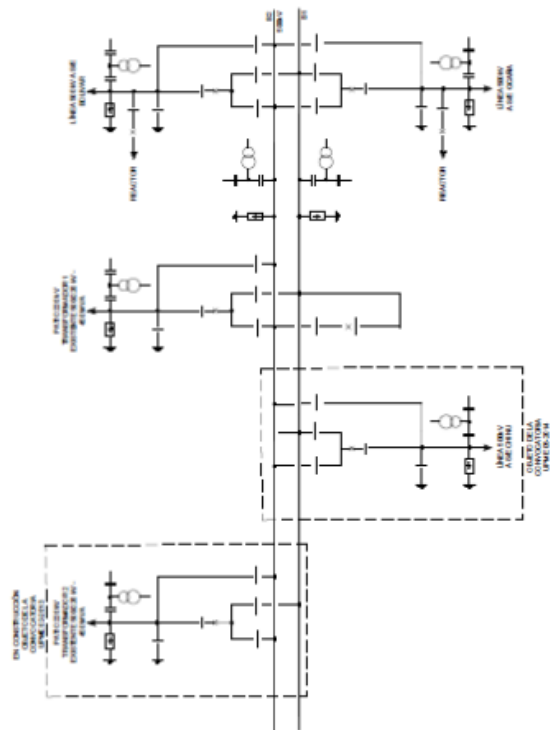
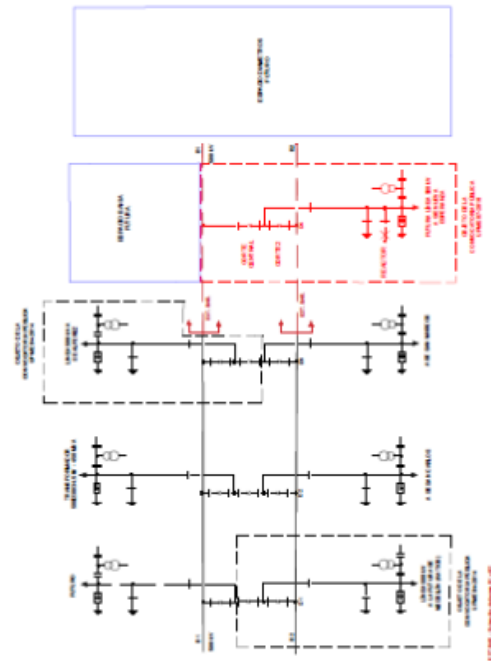


FIGURA 5

FIGURA 2
ESQUEMA UNIFILAR
SUBESTACIÓN LA VIRGINIA 500 kV

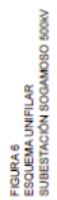
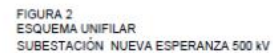
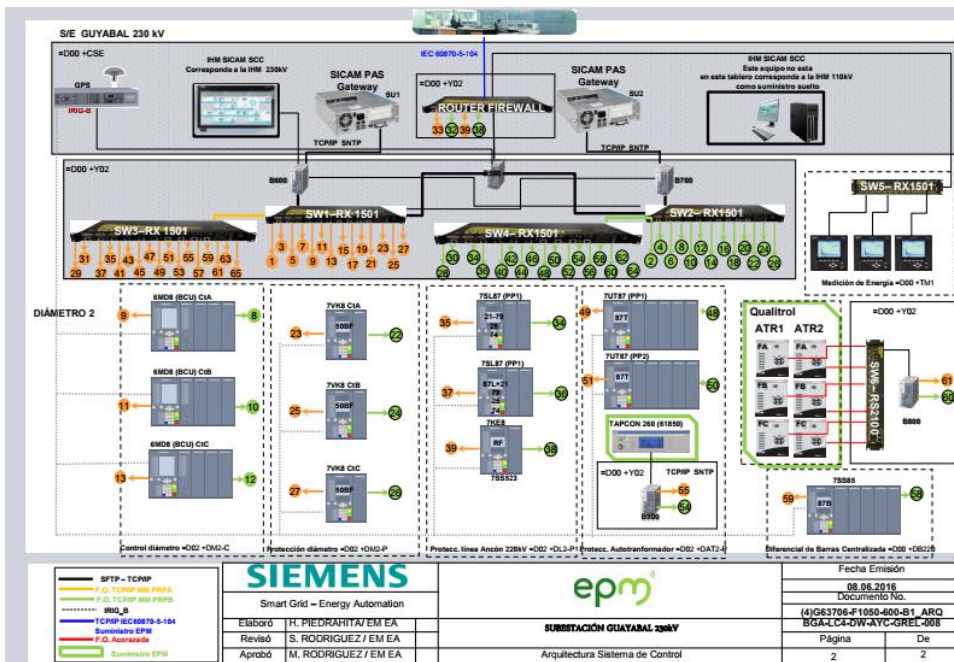
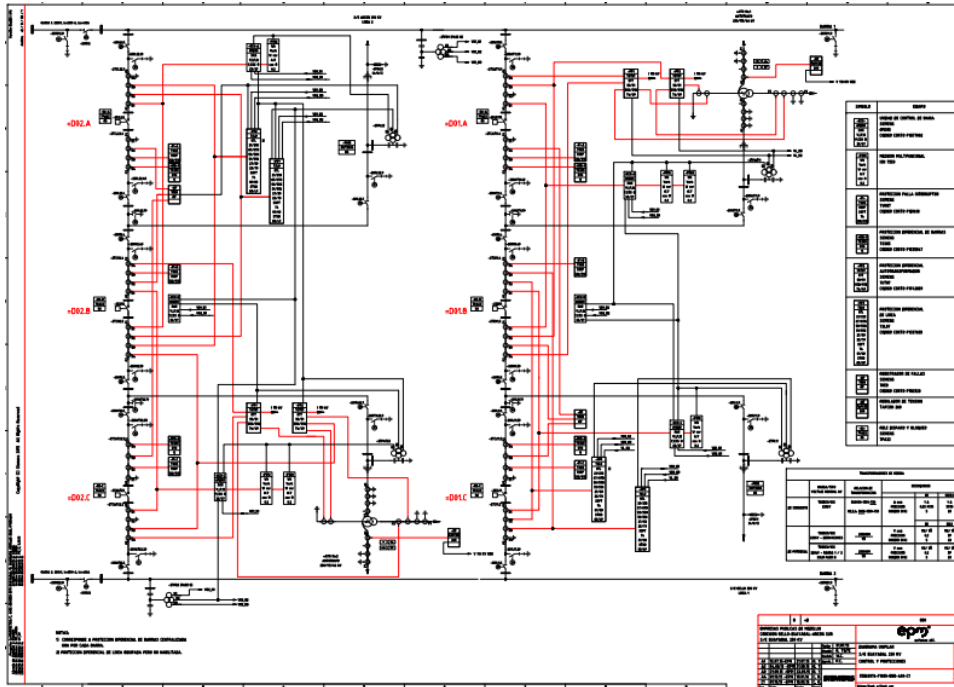
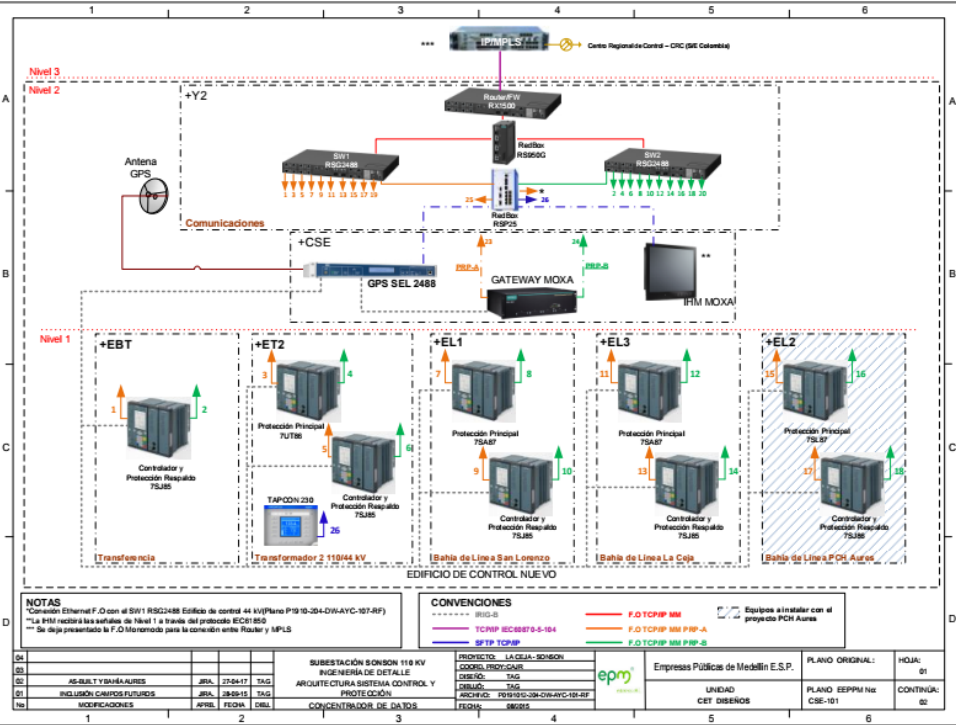
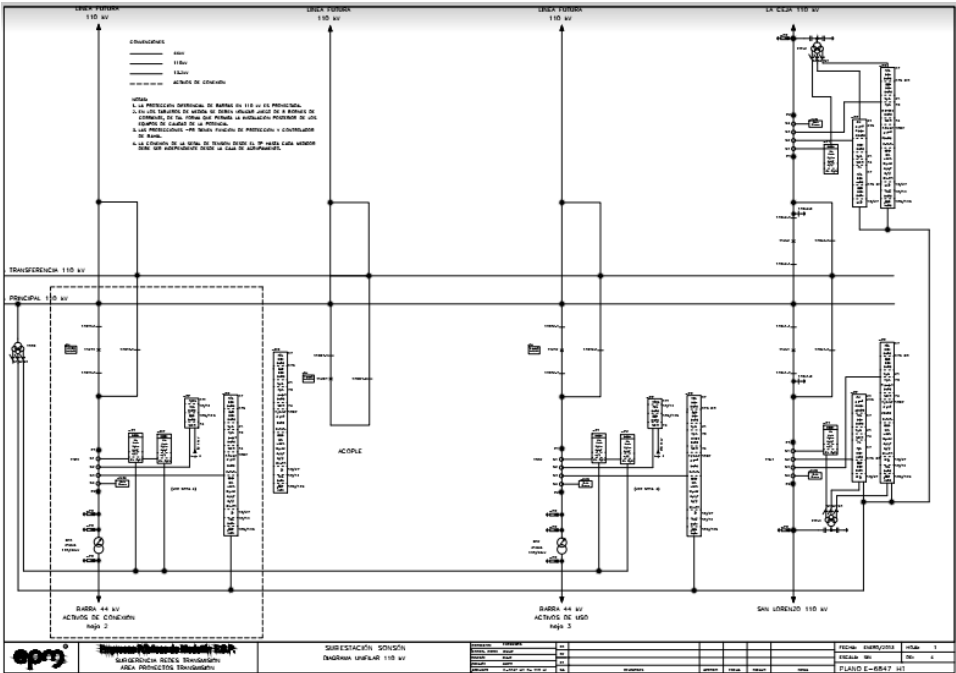


FIGURA 5
ESQUEMA UNIFILAR
SUBESTACIÓN PORCE III

Información suministrada por EPM en la entrevista





Anexo C: Guía para construcción de arquitecturas de bus de proceso en OMNET++

OMNET++ es un marco para la simulación de redes de eventos discretos, orientado a objetos, con una arquitectura genérica que permite su aplicación en el modelado de redes de comunicación, protocolos y colas (Varga & OpenSim Ltd., 2016a). OMNET++ no es un simulador de redes de comunicación, sino más bien una plataforma para desarrollar simulaciones de cualquier sistema que pueda modelarse mediante Eventos Discretos. Es, hasta cierto punto, análogo a una aplicación genérica de programación o modelado de ingeniería, aunque sí está muy orientado a sistemas de eventos discretos que puedan interactuar mediante el envío de mensajes. OMNET++ está basado en C++ y es gratuito para uso académico, aunque existe una versión paga para uso comercial, llamada OMNEST, la cual es licenciada por la empresa SimulCraft Inc (Varga, 2003).

Instalación de OMNET++

OMNET++ se puede ejecutar sobre Windows 7 y superior, Mac OS X 10.10 o superior, Linux, Ubuntu, Fedora, Red Hat, Open SUSE y Generic Unix (Varga & OpenSim Ltd., 2016b).

Para su instalación deben ejecutarse los siguientes pasos:

- Descargar el código fuente gratuitamente, desde <http://omnetpp.org>
- Extraer el archivo .zip en un directorio cuya trayectoria no contenga ningún espacio
- Ejecutar el archivo mingwenv.cmd. este archivo abre una interfaz de consola mediante la cual se puede configurar la plataforma y posteriormente lanzar el entorno de simulación. Véase Figura 1.
- Escribir en consola el comando `config.user`, para verificar los contenidos
- Escribir en consola el siguiente comando, para crear y depurar las librerías

```
$ ./configure
$ make
```
- Ejecutar el ambiente de simulación mediante el comando `$ omnetpp`

Una vez ejecutados estos pasos, el software carga su interfaz gráfica, como se ve en la Figura 2.

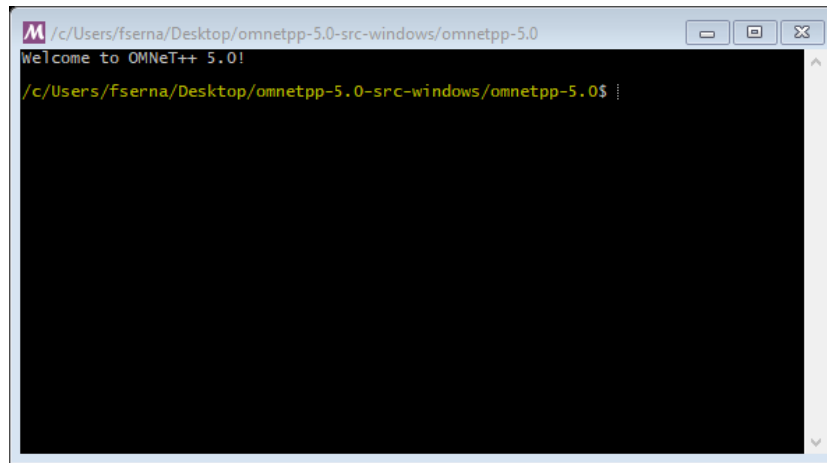


Figura 1. Interfaz de consola de OMNET++

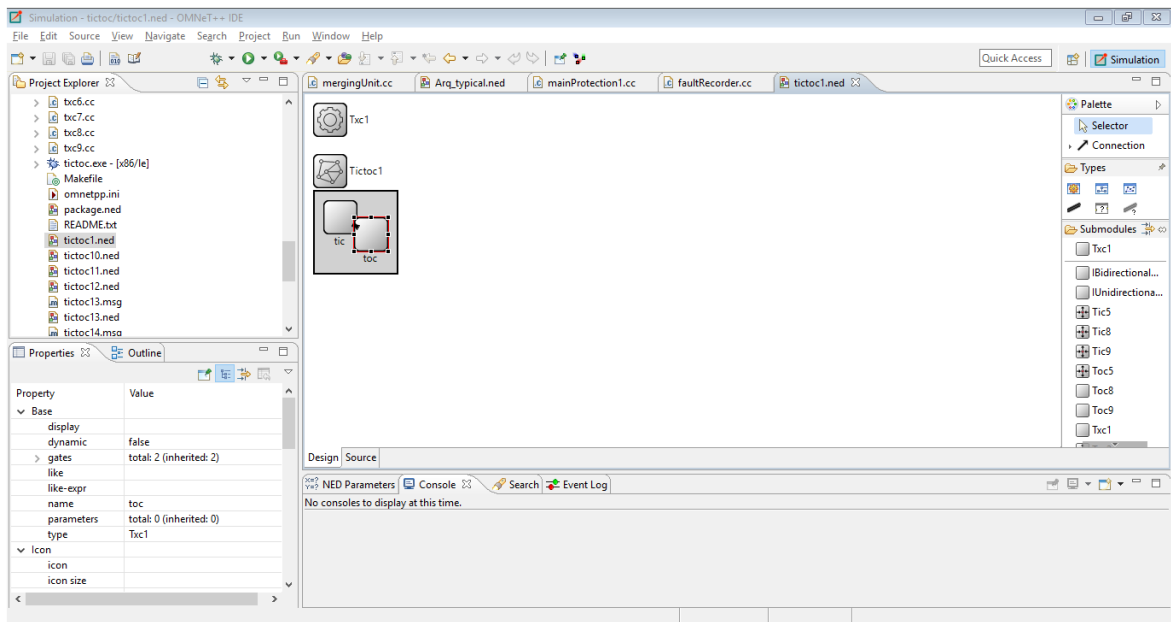


Figura 2. Interfaz gráfica de la plataforma de simulación OMNET++

Plataforma de desarrollo

OMNET++ es una herramienta basada en Eclipse y todo trabajo que se ejecute debe estar contenido en un proyecto, razón por la cual toda simulación en OMNET++ inicia con la creación de un proyecto (Varga & OpenSim Ltd., 2016c).

Para crear un proyecto, se ejecuta la interfaz gráfica de OMNET++, se hace clic en File/New/OMNET++ Project, con lo cual se abre un cuadro de diálogo como el de la Figura 3.

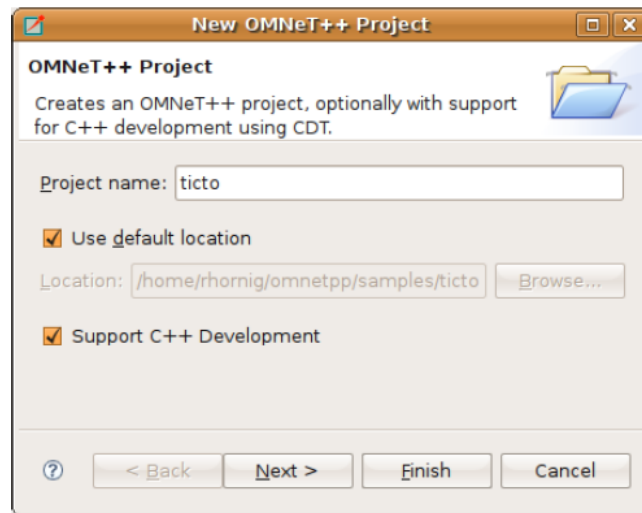


Figura 3. Creación de un proyecto en OMNET++

Una vez creado el proyecto, se puede acceder al directorio generado y comenzar a describir cada uno de sus componentes. Un proyecto puede contener los siguientes componentes:

- Archivos .ned: estos archivos se usan para describir la red objeto de la simulación. Contiene los dispositivos que componen la red y su interconexión. Es un archivo con parte gráfica y parte de código, como se puede apreciar en la Figura 4 y la Figura 5.

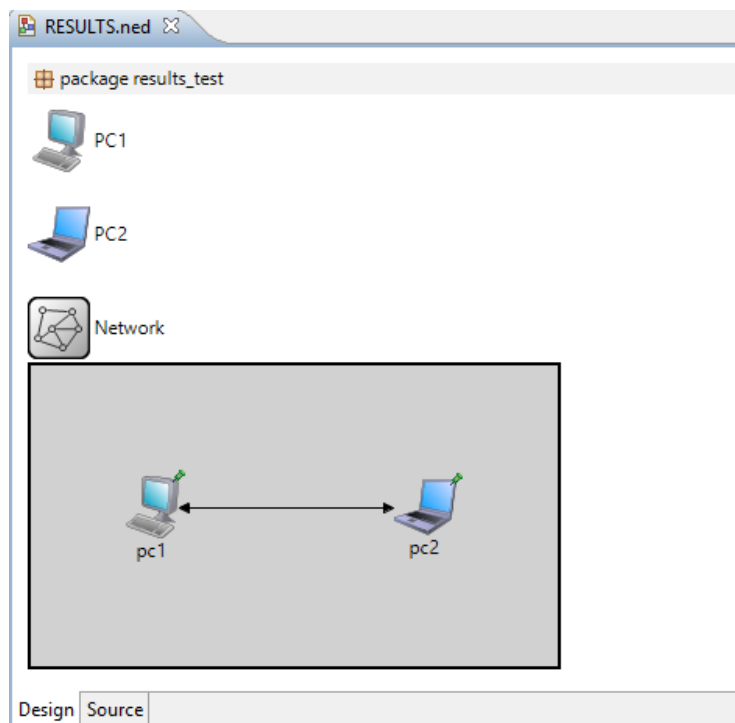


Figura 4. Parte gráfica del archivo .ned

```
package results_test;
//
// TODO documentation
//
simple PC1
{
    parameters:
    @display("i=device/pc");
    @signal[receive](type="long");
    @signal[send](type="long"); //señal para manipular en el archivo .cc Alimenta la señal de
    @statistic[receivedFrames](source="sum(receive)"; record=last; title="received frames");
    @statistic[sendFrames](source="sum(send)"; record=last; title="sent frames");//Señal alimentad

    string mensaje @prompt("Defina el tipo de mensaje: largo o corto");

    gates:
    input PC1_IN;
    output PC1_OUT;
}

//
// TODO documentation
//
```

Figura 5. Parte de código del archivo .ned

- Archivos .cc: estos archivos corresponden al código que describe la inicialización y el comportamiento de cada uno de los componentes de la red, que hayan sido descritos en el archivo .ned. Es necesario crear un archivo .cc por cada componente de la red. Generalmente, la inicialización del

componente se controla con la función `initialize()`. Por su parte, el comportamiento del módulo se define con la función `handleMessage(cMessage *msg)`, en la cual se debe describir qué hace el módulo cuando recibe un mensaje.

- Archivos `.h`: son archivos de descripción típicos de C++, en los cuales pueden definirse macros, nombres, constantes, etc.
- Archivos `.ini`: son archivos que permiten definir el estado inicial de la red. Así como la función `initialize()` permite inicializar cada componente de la red, el archivo `.ini` define el estado inicial de la red misma, definiendo parámetros como retardos, número de conexiones o velocidades de transmisión. También permite consultar al usuario de la simulación los valores de estos parámetros.
- Archivos `.msg`: son archivos que permiten definir estructuras de mensajes personalizadas. OMNET++ trabaja por defecto con objetos tipo `cMessage`, para describir los mensajes, pero el usuario puede crear sus propios mensajes, editando su contenido, sus campos y la forma cómo son gestionados, mediante la creación de archivos `.msg`.
- Archivos `.anf`: estos archivos son creados automáticamente por OMNET++ cuando se ejecuta una simulación, y corresponden al almacenamiento de los resultados de la misma. Para ello, es necesario definir previamente en los archivos `.ned` y `.cc` las señales y las variables a observar durante la simulación.

Construcción de redes en OMNET++

A continuación se describe el procedimiento que debe seguirse para crear un proyecto en OMNET++, de tal forma que la plataforma pueda reconocer los archivos y ejecutar simulaciones.

Creación de un proyecto nuevo

1. Estando en la interfaz gráfica, se hace clic en File/ New/OMNeT++ Project. Se abre un cuadro de diálogo en el que se puede ingresar el nombre del nuevo proyecto. Con esto, OMNET++ crea una carpeta con la información del proyecto.
2. Clic derecho sobre la carpeta del proyecto creado y luego clic en New / Network Description File (NED).
3. Doble clic sobre el archivo `.ned` para abrirlo en la consola. Debe visualizarse una ventana con dos pestañas: Design y Source. La pestaña Design muestra la gráfica

de la red y sus componentes y la pestaña Source contiene el código de configuración de la red. Sobre el lado derecho de la consola se encuentra la ventana Types. Sobre esta ventana se selecciona “simple module, create simple module” y se hace clic en el área de trabajo. Se agregan tantos módulos como sean necesarios.

4. En la ventana Types, se selecciona “Network – Create a network type” y se hace clic en el área de trabajo
5. Se arrastran los módulos antes creados en el paso 3 hacia el módulo Network
6. Antes de interconectar los módulos en la red, es necesario definir los puertos de entrada y salida de cada módulo. Para ello se hace clic en la pestaña Source del archivo .ned e incluir el siguiente código, agregando y cambiando los puertos deseados.

```
simple Module
{
    parameters:

    gates:
        input IN_PORT_NAME;
        output OUT_PORT_NAME;
}
```

7. Volver a la pestaña Design. En la ventana Palette, se hace clic en Connection. Si se quieren conectar los módulos A y B, se hace clic sobre un módulo A y luego clic sobre el B, seleccionando los puertos correspondientes.

Nota importante: El simulador no correrá si no se han efectuado todas las conexiones, no se pueden dejar puertos al aire, sin conexión.

8. Una vez se tenga lista la conexión de todos los módulos, se procede a definir su lógica
9. Clic derecho sobre la carpeta del proyecto / New / Source File. Se elige un nombre y debe terminar con la extensión .cc

Nota importante: Es necesario crear los archivos .cc de todos los módulos (así estén vacíos y no se describa nada en estos archivos) antes de correr la primera simulación. El simulador no correrá a menos que encuentre la definición de todos los módulos con la instrucción Define_Module(nombre del módulo); es sensible a mayúsculas y minúsculas,

de manera que lo que se pone entre paréntesis debe ser exactamente el mismo nombre que se le dio al módulo en el punto 3.

10. Se crea un archivo de inicialización. Clic derecho sobre la carpeta del proyecto / New / Initialization File. Se abre este archivo y se cambia el nombre por el de la red que se agregó en el paso 4.

Envío de mensajes

Los mensajes son programados al interior de los archivos .cc, utilizando las funciones `scheduleAt`(time, *msg); `send`(*msg, OutPort); y `SendDelayed` (time, *msg, OutPort);

Primero se debe inicializar alguno de los módulos con un automensaje, para iniciar el tráfico. Los automensajes no se muestran en consola como eventos, pero sí se cuentan internamente.

```
void MergingUnit1::initialize()
{
    cMessage *start = new cMessage("SV_Start");//mensaje para iniciar el
    envío de SV
    scheduleAt(0.0, start);
}
```

cMessage sólo permite mensajes argumentos tipo Char.

Creación de paquetes

Debido a que cMessage sólo admite argumentos Char, si se quieren manejar diversas variables al interior de un mensaje es necesario concatenar cadenas de caracteres en el nombre del mensaje. Esto es útil por ejemplo para agregar direcciones fuente y destino, bits de calidad u otra información. La concatenación de las variables en el nombre puede hacerse de la siguiente forma:

```
std::string Source_Address;
std::string Destination_Address;
std::string Info;

Info = Source_Address + Destination_Address;

char Sample[1024];
strcpy(Sample, SV.c_str());

cMessage *start = new cMessage(Sample);
scheduleAt(0.0, start);
```

Sin embargo, este procedimiento puede ser muy engorroso para mensajes que contengan tramas más complejas. Por esta razón, se recomienda crear paquetes específicos para cada protocolo o estructura de trama.

El procedimiento para la creación de paquetes es el siguiente:

1. Clic sobre la carpeta del proyecto / New / Message Definition (msg)
2. Se define el paquete con sus componentes. El siguiente código es un ejemplo de los componentes de un paquete *Sampled Value*, los cuales incuyen los bits de calidad y sincronización, así como las corrientes y voltajes de cada fase.

```
packet SampledValue{  
    int QualityBIT;  
    int SynchBIT;  
    int Sequence;  
    int VPH1;  
    int VPH2;  
    int VPH3;  
    int NV;  
    int IPH1;  
    int IPH1;  
    int IPH1;  
    int NI;  
}
```

3. CTRL+B o Project / Build All. Esto genera los archivos XXXX_m.cc y XXXX_m.h a partir del archivo XXXX.msg, donde XXXX es el nombre dado al archivo .msg en el paso 1.
4. En el archivo .cc del módulo que va a emplear el paquete se escribe #include <XXXX_m.h>

Definición y uso de variables aleatorias

OMNET++ cuenta con una librería de funciones aleatorias, las cuales permiten generar variaciones aleatorias en determinadas partes de la simulación. La Tabla 1 muestra las funciones aleatorias de OMNET++ (Varga & OpenSim Ltd., 2016a).

Tabla 1. Listado de funciones aleatorias de OMNET++

Distribution	Description
Continuous distributions	
<i>uniform</i> (<i>a, b</i>)	uniform distribution in the range [a,b]
<i>exponential</i> (<i>mean</i>)	exponential distribution with the given mean
<i>normal</i> (<i>mean, stddev</i>)	normal distribution with the given mean and standard deviation
<i>truncnormal</i> (<i>mean, stddev</i>)	normal distribution truncated to nonnegative values
<i>gamma_d</i> (<i>alpha, beta</i>)	gamma distribution with parameters $\alpha > 0$, $\beta > 0$
<i>beta</i> (<i>alpha1, alpha2</i>)	beta distribution with parameters $\alpha_1 > 0$, $\alpha_2 > 0$
<i>erlang_k</i> (<i>k, mean</i>)	Erlang distribution with $k > 0$ phases and the given mean
<i>chi_square</i> (<i>k</i>)	chi-square distribution with $k > 0$ degrees of freedom
<i>student_t</i> (<i>i</i>)	student-t distribution with $i > 0$ degrees of freedom
<i>cauchy</i> (<i>a, b</i>)	Cauchy distribution with parameters a, b where $b > 0$
<i>triang</i> (<i>a, b, c</i>)	triangular distribution with parameters $a \leq b \leq c$, $a \neq c$
<i>lognormal</i> (<i>m, s</i>)	lognormal distribution with mean m and variance $s > 0$
<i>weibull</i> (<i>a, b</i>)	Weibull distribution with parameters $a > 0$, $b > 0$
<i>pareto_shifted</i> (<i>a, b, c</i>)	generalized Pareto distribution with parameters a, b and shift c
Discrete distributions	
<i>intuniform</i> (<i>a, b</i>)	uniform integer from $a..b$
<i>bernoulli</i> (<i>p</i>)	result of a Bernoulli trial with probability $0 \leq p \leq 1$ (1 with probability p and 0 with probability $(1-p)$)
<i>binomial</i> (<i>n, p</i>)	binomial distribution with parameters $n \geq 0$ and $0 \leq p \leq 1$
<i>geometric</i> (<i>p</i>)	geometric distribution with parameter $0 \leq p \leq 1$
<i>negbinomial</i> (<i>n, p</i>)	negative binomial distribution with parameters $n > 0$ and $0 \leq p \leq 1$
<i>poisson</i> (<i>lambda</i>)	Poisson distribution with parameter λ

Las funciones aleatorias son muy útiles para dar realismo a una simulación en OMNET++. Por ejemplo, pueden emplearse para generar mensajes a intervalos de tiempo aleatorios:

```
scheduleAt(simTime() + exponential(1.0), msg);
```

Otra aplicación muy útil de las funciones aleatorias es la definición de canales de conexión que presenten retardos aleatorios o, más importantes aún, modelen el ruido mediante funciones de distribución aleatorias:

```
a.g++ <--> {delay=10ms; ber=geometric(1e-8);} <--> b.g++;
```

Registro de información en las simulaciones

OMNET++ permite guardar información sobre los paquetes enviados, su estampa de tiempo, los saltos que realizó cada mensaje, la recepción exitosa o fallida de mensajes, los retardos, entre muchas otras variables. Todo esto se realiza a partir de la declaración de estadísticas y señales que se activan al enviar o recibir mensajes, como se describe a continuación.

Declaración de estadísticas

Las estadísticas se definen en el archivo .ned, como una combinación de una señal y una estadística en cada módulo de la red. Por ejemplo, el siguiente código define una estadística para llevar el conteo de paquetes en cola para el módulo Queue, mediante el registro de la estadística queueLength, la cual se activará posteriormente en el archivo .cc invocando la señal qlen

```
simple Queue
{
  parameters:
    @signal[qlen](type=int); // optional
    @statistic[queueLength](source=qlen; record=max,timeavg,vector?);
  ...
}
```

Como se puede apreciar en el ejemplo anterior, las estadísticas cuentan con parámetros como la source y record. Source indica cuál señal alimenta el registro de la estadística y por lo general se asocia al nombre de la señal correspondiente (qlen en este caso). Record por su parte puede configurarse a partir de filtros y registros predefinidos por OMNET++. Por ejemplo, max registra el mayor valor obtenido durante la simulación, timeavg registra el promedio de los tiempos de ocurrencia de eventos, etc.

El listado completo de filtros y registros de estadísticas se muestra en la Tabla 2 y la Tabla 3.

Tabla 2. Listado de filtros de estadísticas de OMNET++

Filter	Description
count	Computes and outputs the count of values received so far.
sum	Computes and outputs the sum of values received so far.
min	Computes and outputs the minimum of values received so far.
max	Computes and outputs the maximum of values received so far.
mean	Computes and outputs the average (sum / count) of values received so far.
timeavg	Regards the input values and their timestamps as a step function (sample-hold style), and computes and outputs its time average (integral divided by duration).
constant0	Outputs a constant 0 for each received value (independent of the value).
constant1	Outputs a constant 1 for each received value (independent of the value).
packetBits	Expects cPacket pointers as value, and outputs the bit length for each received one. Non-cPacket values are ignored.
packetBytes	Expects cPacket pointers as value, and outputs the byte length for each received one. Non-cPacket values are ignored.
sumPerDuration	For each value, computes the sum of values received so far, divides it by the duration, and outputs the result.
removeRepeats	Removes repeated values, i.e. discards values that are the same as the previous value.

Tabla 3. Listado de registros de estadísticas de OMNET++

Recorder	Description
last	Records the last value into an output scalar.
count	Records the count of the input values into an output scalar; functionally equivalent to <code>last(count)</code>
sum	Records the sum of the input values into an output scalar (or zero if there was none); functionally equivalent to <code>last(sum)</code>
min	Records the minimum of the input values into an output scalar (or positive infinity if there was none); functionally equivalent to <code>last(min)</code>
max	Records the maximum of the input values into an output scalar (or negative infinity if there was none); functionally equivalent to <code>last(max)</code>
mean	Records the mean of the input values into an output scalar (or NaN if there was none); functionally equivalent to <code>last(mean)</code>
timeavg	Regards the input values with their timestamps as a step function (sample-hold style), and records the time average of the input values into an output scalar; functionally equivalent to <code>last(timeavg)</code>
stats	Computes basic statistics (count, mean, std.dev, min, max) from the input values, and records them into the output scalar file as a statistic object.
histogram	Computes a histogram and basic statistics (count, mean, std.dev, min, max) from the input values, and records the result into the output scalar file as a histogram object.
vector	Records the input values with their timestamps into an output vector.

Cuando se han definido estadísticas y se ejecuta una simulación, OMNET++ crea automáticamente un archivo de extensión .anf, en la carpeta results del proyecto. Este archivo contiene otros archivos, como el .sca para el registro de escalares y el .vec para el registro de vectores.

El archivo .anf permite crear conjuntos de datos (datasets) para luego graficar las estadísticas registradas. Para ello, se hace clic derecho sobre el archivo .anf y luego clic en plot.

Para crear un dataset se seleccionan los datos en el archivo .anf que se quieren incluir en el conjunto; luego se hace clic derecho sobre los datos seleccionados y se selecciona “Add to Dataset”. El resultado se visualiza en la pestaña Datasets, como en la Figura 6.

Por ejemplo, se puede hacer un dataset con las estadísticas de mensajes generados y otro con las estadísticas de los mensajes recibidos. Luego, en la pestaña Datasets, se busca al lado derecho las diferentes opciones de visualización de resultado, como diagramas de barras, histogramas o diagramas de dispersión.

Datasets and Charts

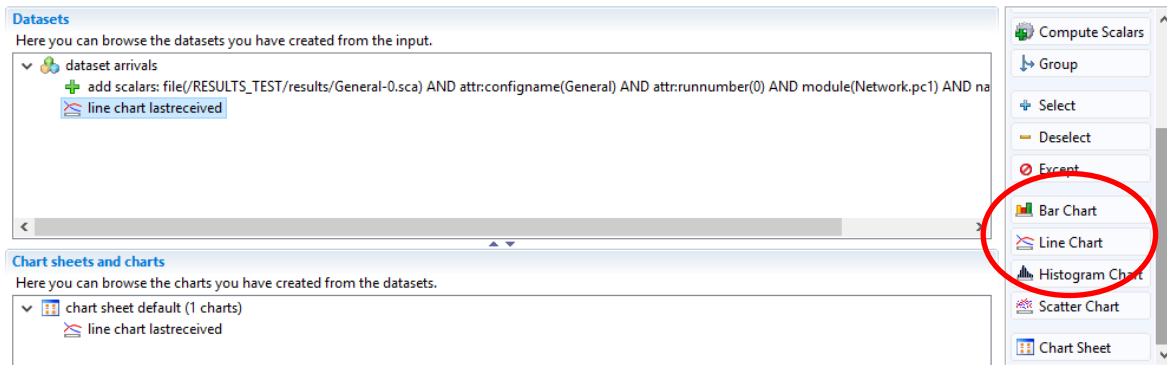


Figura 6. Interfaz para creación de gráficos de estadísticas

Procedimiento para efectuar el conteo de paquetes:

En el archivo .ned:

Es necesario definir dos señales para contar los paquetes enviados o recibidos. Esto se hace en cada módulo sobre el que se requiera guardar algún resultado o estadística:

```
simple PC1
{
  parameters:

    @signal[sent](type="long"); //señal para manipular en el archivo .cc Alimenta la
    señal de conteo real

    @statistic[sentFrames](source="sent"; record=last; title="sent frames");

  gates:
    input PC1_IN;
    output PC1_OUT;
}
```

Si se emplea "last", contará el número de paquetes enviados. Si se usa "sum", sumará el consecutivo de los paquetes enviados: por ejemplo si se envían 4 paquetes, last producirá un resultado igual a 4, mientras que sum producirá $1+2+3+4=10$

En el archivo .cc

Es necesario manipular una de las señales creadas en el archivo .cc del módulo sobre el cual se requieren las estadísticas.

Se crea una nueva variable para registrar a la señal a modificar, además de un contador a incrementar:

```
class PC1 : public cSimpleModule
{
    protected:

        simsignal_t sentSignal;
        long sentCounter;

        virtual void initialize() override;
        virtual void handleMessage(cMessage *msg) override;
        virtual void finish() override;
};
```

En la inicialización, se registra la señal a modificar:

```
void PC1::initialize()
{
    sentCounter=0;
    sentSignal = registerSignal("sent");
}
```

Durante el envío de un mensaje, se incrementa el contador y se emite la señal. Si la señal no es emitida, no se guardarán los valores en la otra señal definida en el archivo .ned y no se obtendrán resultados:

```
void PC1::handleMessage(cMessage *msg)
{
    send(msg, "PC1_OUT");

    emit(sentSignal, ++sentCounter);
}
```

Procedimiento para contar retardos entre mensajes

No es necesario crear señales. En el archivo .cc del módulo sobre el cual se quiere medir el retardo, se crea un vector para almacenar los tiempos de retardo:

```
class PC1 : public cSimpleModule
{
    protected:
        cOutVector endToEndDelayVec;
```

```

    virtual void initialize() override;
    virtual void handleMessage(cMessage *msg) override;
    virtual void finish() override;
};

```

En la función de inicialización, se asigna un nombre al vector

```

void PC1::initialize()
{
    endToEndDelayVec.setName("End-to-End Delay");
}

```

Al recibir un mensaje, se almacena el valor del retardo, como la diferencia entre el tiempo actual de simulación y el instante en que fue creado el mensaje:

```

void PC1::handleMessage(cMessage *msg)
{
    simtime_t eed = simTime() - msg->getCreationTime();
    endToEndDelayVec.record(eed);
    delete msg;
}

```

Si se quiere medir sólo retardos desde el último envío del mensaje, se debe destruir el mensaje al final. En caso de requerirse enviar el mensaje a otro módulo, se debe crear uno nuevo como copia del recibido.

Retardo por una ruta entre varios módulos:

Si no se pone la instrucción `delete msg`, y el mensaje se sigue enviando a otros módulos, el valor de `msg->getCreationTime()` será siempre el mismo y lo que se medirá será el tiempo empleado por el mensaje en todo su recorrido por varios módulos.

Ingreso de parámetros por parte del usuario

Cada módulo de la red puede contener parámetros a definir en el momento de lanzar la simulación. Esto es útil por ejemplo para definir el número de clientes en una conexión, modelar diferentes retardos o para interactuar con el usuario.

Cada parámetro debe estar asignado a su respectivo módulo. Para ello, en el archivo .ini, se hace referencia al módulo y su correspondiente parámetro, por el cual se va a preguntar al usuario:

```
[General]
network = Network

[Config User]
description = "Pregunta por el retardo en el canal"
Network.retardo=ask
```

De esta manera, el parámetro retardo es asignado al módulo Network en el archivo .ini.

Luego, es necesario usar dicho parámetro al interior del módulo correspondiente, en el archivo .ned

Como se muestra a continuación, el parámetro puede tener varios atributos: @prompt permite imprimir una cadena para explicar al usuario qué debe ingresar; @unit permite definir las unidades del parámetro.

```
network Network
{
    parameters:
        double retardo @prompt("Retardo en el canal: ingrese el tiempo
que tarda y finalice con la letra s (segundos)") @unit(s); // @unit(s) es
necesario, sino no convierte a segundos
    submodules:
        pc1: PC1 {
            @display("p=78,91");
        }
        pc2: PC2 {
            @display("p=253,91");
        }
    connections:
        pc1.PC1_OUT --> {delay = retardo; datarate=1bps; ber=1e-10;} -->
pc2.PC2_IN;
        pc2.PC2_OUT --> {delay = 50ms; datarate=1bps; ber=1e-10;} -->
pc1.PC1_IN;
}
```

Para más información, revisar los archivos ClientServer.ned y omnetpp.ini del ejemplo dyna.

Conexión de múltiples dispositivos a pedido del usuario

Esta técnica puede ser útil para modificar la topología de la red a voluntad del usuario, por ejemplo definiendo un número mayor o menor de clientes o servidores dependiendo de las dimensiones deseadas de la red.

Para configurar un número variable de dispositivos en la red, se define como un parámetro en el archivo .ini, de la siguiente manera:

```
[Config User]
description = "ask number of client computers"
ClientServer.numClients = ask
```

Adicionalmente, en el archivo .ned se utiliza este parámetro para definir el número de conexiones. Las conexiones se realizan mediante vectores, cuya dimensión es el número de clientes definido en el archivo .ini

```
network ClientServer
{
    parameters:
        int numClients @prompt("Number of IEDs") = default(1);
    submodules:
        server: Server {
            parameters:
                @display("p=210,70");
        }
        switch: Switch {
            parameters:
                pkRate = 1.5*numClients; // pkRate should be >=
numClients, otherwise switch will become the bottleneck
                queueMaxLen = 20; // buffer max 20 packets
                @display("p=210,170");
            gates:
                port[numClients+1];
        }
        ied[numClients]: IED {
            parameters:
                timeout = 5s;
                @display("p=70,270,m,10,80");
        }
    connections:
        for i=0..numClients-1 {
            ied[i].port <--> { delay = 10ms; } <--> switch.port[i];
        }
}
```

```
server.port <--> { delay = 10ms; } <-->  
switch.port[numClients];  
}
```

Marcos externos

Debido a que OMNET++ es un entorno de desarrollo general, no cuenta por sí solo con módulos prediseñados para emular protocolos reales de comunicación, como por ejemplo TCP-IP o UDP. No obstante, siendo OMNET++ una herramienta abierta, terceros han desarrollado una variedad de “marcos externos” para implementar protocolos concretos, facilitando en parte el trabajo de programación mediante el suministro de “plantillas” de paquetes o módulos de red.

Algunos de los marcos externos más relevantes son los siguientes:

- INET: es una librería de código abierto, que contiene protocolos y modelos de redes inalámbricas y móviles. Incluye plantillas de protocolos de internet como TCP, UDP, IPv4, IPv6, OSPF y BGP. Incluye también paquetes útiles para modelar protocolos Ethernet y el estándar IEEE 802.11 (INET Team, 2016). INET es concebido como un proyecto colaborativo, se alimenta de los aportes de los usuarios que hacen parte de una comunidad, en la cual cada uno aporta sus códigos y desarrollos. Por esta razón, es un marco en constante desarrollo; su manual de uso se encuentra en borrador.
- INETMANET: es un marco orientado a la simulación de redes móviles Ad Hoc. Una red MANET (*Mobile Ad-hoc Network*) es una red inalámbrica sin infraestructura fija de acceso, en la cual los dispositivos pueden moverse físicamente y conectarse y desconectarse dinámicamente a otros dispositivos que estén cerca, de manera que cada nodo del sistema tiene la capacidad de actuar como enrutador en determinados momentos (Conti & Giordano, 2014). INETMAET es el marco para OMNET++ que contiene modelos para redes Ad-hoc, tales como nodos móviles y control del canal (Noorhayati, Norashidah, Ejaz, & Awang, 2016). La librería puede ser descargada de la página web: <https://github.com/aarizaa/inetmanet-3.x>
- SimuLTE: las redes *Long Term Evolution* – *LTE* son un estándar UMTS (3GPP-TS 36.300) para redes celulares. SimuLTE es un marco para simulación de redes LTE, desarrollado por el grupo de investigación *Computer Networking Group*, de la Universidad de Pisa.

SimuLTE implementa los datos de las redes de acceso LTE y el núcleo de paquetes empleados por estas redes (Virdis, Stea, & Nardini, 2014). Las librerías de este marco pueden descargarse en la página web <http://simulte.com/index.html>

Estos marcos externos pueden ser muy útiles para la creación rápida de redes complejas, ya que contienen plantillas de protocolos como el IEEE 802.11, tramas Ethernet de protocolos como TCP-IP, TCP- UDP, entre muchas otras funcionalidades predefinidas.

Referencias bibliográficas

- Conti, M., & Giordano, S. (2014). Mobile ad hoc networking: milestones, challenges, and new research directions. *IEEE Communications Magazine*, 52(1), 85–96.
- INET Team. (2016). INET Framework for OMNeT++. Retrieved from <https://omnetpp.org/doc/inet/api-current/inet-manual-draft.pdf>
- Noorhayati, M. N., Norashidah, M. D., Ejaz, A., & Awang, N. (2016). OMNET++ based cognitive radio simulation network. *2016 IEEE 7th Control and System Graduate Research Colloquium (ICSGRC 2016)*.
- Varga, A. (2003). OMNET++ Academic Public License. Retrieved from <https://omnetpp.org/intro/license>
- Varga, A., & OpenSim Ltd. (2016a). OMNeT++ Simulation Manual. Version 5.2. Retrieved from <https://omnetpp.org/doc/omnetpp/SimulationManual.pdf>

Varga, A., & OpenSim Ltd. (2016b). OMNeT++ Installation Guide. Retrieved from <https://omnetpp.org/doc/omnetpp/InstallGuide.pdf>

Varga, A., & OpenSim Ltd. (2016c). OMNeT++ User Guide. Retrieved from <https://omnetpp.org/doc/omnetpp/UserGuide.pdf>

Viridis, A., Stea, G., & Nardini, G. (2014). SimuLTE - A modular system-level simulator for LTE/LTE-A networks based on OMNeT++. *2014 4th International Conference On Simulation And Modeling Methodologies, Technologies And Applications (SIMULTECH)*.